

NOS/VE 1.2.1 L670

SOFTWARE RELEASE BULLETIN

Control Data Corporation recommends that the Software Release Bulletin be read in its entirety prior to any installation.

SMD131271

Contents

SRB Introduction	2
Audience	2
Organization	2
Documented But Not Supported Products or Features	2
Chapter 1 - Installation Overview	3
Significant Problems	3
Deadstarting After Changing Mode or Memory Parameters	3
Avoid Typeahead at CTI Deadstart Time	3
Upgrading NOS/VE 1.2.1 L670 with Job Recovery	4
SETVE Procedure Fails in Secured Mode	4
Error PF 2044 If All Products Not Installed	4
Inconsistent Results from VERIFY_INSTALLED_SOFTWARE	4
Notes and Cautions	5
NOS 2.4.3 Transparent Mode CCP Problem	5
Installation Changes	5
Chapter 2 - Initial Installation	7
NOS/VE Installation Summary	8
Deadstart Information Checklist	9
Step 1 Initiate NOS/VE Deadstart	10
Step 2 Deadstart Parameters	13
Step 3 System Core Commands	15
Step 4 Physical Configuration	16
Step 5 Installing the BASE_1 Tape	17
Step 6 Setup Installation Process	18
Step 7 Defining the NOS/VE Network	19
Defining the Network Configuration	19
Resetting the configuration terminal DI.	21
Step 8 Configuration Terminal	22
Step 9 Define Your Physical Configuration	23
Create the PHYSICAL_CONFIGURATION file.	23
Step 10 Create a New Deadstart Tape	26
Step 11 Redeadstart the System	28
Step 12 Install the Products	30
Step 13 Copy the Deadstart File to Disk	31
Step 14 Automatic Deadstart	32
After You Have Installed NOS/VE	34
Chapter 3 - Upgrade Installation of NOS/VE	35
Step 1 Miscellaneous Changes	36
Step 2 Deadstart NOS/VE Level 670	37
Step 3 System Core	38
Step 4 Installation Option Menu	39
Step 5 Installing the BASE_1 Tape	40
Step 6 Installing the Products	41
After You Have Upgraded NOS/VE	42

Chapter 4 - NOS/VE Product Installation	43
Standard and Optional Product Installation	43
Network Installation	45
Installation of PTF, SCF, and SCFS	46
SCF and SCFS Installation	46
PTF Installation	48
Chapter 5 - Installing a Correction Package	50
Correction Package Installation Process	51
Chapter 6 - Dual-State Preparation	53
Step 1 Generate NOS/VE Deadstart Procedure Files	53
Step 2 Reserve VE Elements (NOS only)	53
Step 3 Enable the NVE Subsystem (NOS only)	54
Chapter 7 - Accounting	55
Installation	55
Notes and Cautions	55
Ignore ERROR Issued by INSTALL_ACCOUNTING_ANALYSIS	55
Manual Errata	55
Counters Raised from 63 to 256	56
Chapter 8 - Advanced Access Methods (AAM)	57
Significant Problems	57
Sharing Files on Dual Processor Systems	57
Certain File Attributes Not Supported	57
Chapter 9 - APL	58
Notes and Cautions	58
END_PARTIAL_CHARACTER Attribute Reset by APL	58
Conversion Utility	58
Manual Errata	58
CCP to CDCNET Differences with APL	59
Chapter 10 - BASIC	61
Significant Problems	61
Symbolic Debugging Using SET_STEP_MODE	61
Notes and Cautions	61
Do Not Use Concurrent OPENS	61
Chapter 11 - Basic Access Methods (BAM)	62
Significant Problems	62
PRESET_VALUE File Attribute	62
MINIMUM_RECORD_LENGTH File Attribute	62
Reporting of Boundary Conditions	62
PSP\$OPEN_FILE Opens File Regardless of Error Condition	63
SET_FILE_ATTRIBUTE Sets Some Attributes Incorrectly	63
Files Implicitly Attached Are Not Implicitly Detached	63
File Label Attributes Cannot Be Altered After Creation	63
Notes and Cautions	63
Circular Connections with COPY_FILE	64

Chapter 11 - Basic Access Methods (BAM) (Continued)	
Share Mode Messages	64
SET_WORKING_CATALOG Limitations	64
Chapter 12 - CDCNET	66
Installation - NOS/VE Only	66
Initial Installation	66
Upgrade Installation	67
Significant Problems	68
L664 Problems Corrected	68
Dump Analyzer (ANACD) Deficiencies	70
MANCC Deficiencies	71
Network Performance Analyzer (NPA) Deficiencies	72
Log Files in Date/Time Order	72
Automated Command Entry At Initial Connection May Fail.	73
Line Hangs	73
KILL_SYSTEM_DUMP Parameter not Honored	74
MDI/MTI Resets During Host Mainframe Maintenance - NOS/VE Only	74
Mismatching X.25 Network Identifiers can Reset DI	75
No Output From 53X Printer - NOS/VE Only	75
Congested MDIs - NOS and NOS/VE Dual State	75
Unrecognized NETOU Secondary User Statements - NOS Only	76
NOS Login - NOS and NOS/VE Dual State	76
No Disconnect for IAF Inactivity Timeout - NOS Only	76
NETFM Local File Name Change - NOS Only	76
Misleading A-A Connection Failure Reason Codes - NOS Only	77
First Time Installation Without FSE - NOS Only	77
Notes and Cautions	77
Multiple Interactive Connections	77
CREATE_CONNECTION Commands in Process	78
Non-Default Buffer Allocation Parameters	78
Adding X.29 Log Messages to Configuration Procedures	79
DI Configuration Files	79
CIM Configuration Limitations	79
LIM Configuration Caution for Synchronous Lines	79
Rotary Support	80
Condensed Statistics Removed from REFCLF	80
Accessing CDCNET Permanent Files - NOS/VE Only	80
Send_Command (SENC) Parameter Order - NOS Only	81
Procedure TC7TUP - NOS Only	81
CDCNET DI as only Network Hardware - NOS Only	81
63 Character Set Not Supported - NOS Only	82
LCF Control Limitations for Auto Login - NOS Only	82
Operational Considerations	82
NOS-NOS/VE MDI Differences	83
CDCNET R1.1 DI Memory Requirements	83
Access Procedures for ANACD, MANCC and NPA - NOS/VE Only	87
ANACD - Dump Analyzer	88
Flow Control	88
DI Clock Maintenance	90
NPA	91

Chapter 12 - CDCNET (Continued)	
Excessive Logging	91
Recommendations for Logging Configuration	91
NAM/VE Attribute Management Differences	92
Access Path Determination - NOS Only	93
Accounting Information - NOS Only	93
NAM Terminal Name Generation	94
Account File Messages	94
Application/Terminal User Information	95
Host Unavailability Processing	95
Hi-Speed Micro-Mainframe File Transfers	95
Terminal Recognition Problems	95
User Breaks	96
Line Folding Changes	96
CDCNET Batch Devices	96
HASP Workstations	97
CDC 533-1, 536-1 Asynchronous Line Printers	98
Printing of the Banner Page Separator	98
Manual Errata	98
INPUT_OUTPUT_MODE Documentation	99
CDCNET Secure Terminal Procedure	100
HASP Workstation Configuration	100
Logging Configuration Command Changes	100
Log Message Changes	102
Command Continuation	102
Accessing CDCNET Utilities - NOS/VE Only	102
Retrieving a DI Dump File - NOS/VE Only	102
Terminal Class for Teletype Model 43 - NOS & NOS/VE Dual State	103
X.25 Configuration Recommendation - NOS Only	103
MDI Error Status - NOS Only	103
Conversion of Transparent Attributes - NOS & NOS/VE Dual State	103
Effects of Changing Terminal Classes - NOS & NOS/VE Dual State	104
CDCNET Mapping of NAM FN/FV Pairs - NOS & NOS/VE Dual State	105
Effects of NOS TRMDEF Command - NOS and NOS/VE Dual State Only	108
Future NOS Considerations - NOS Only	111
Chapter 13 - COBOL	112
Significant Problems	112
Compilation Hang Using COPY Statement	112
Duplicate User Defined Names Not Diagnosed	112
COBOL Parameter ERROR Does Not Work As Documented	112
Notes and Cautions	113
All COBOL Programs Must Be Recompiled	113
AUDIT Compilation Parameter	113
Interprogram Communication	113
New COBOL Reserved Words	114
COBOL Doesn't Support Multiple Statement Debugging	114
COBOL Parameter BASE_LANGUAGE=ANS85 Has No Effect	115

Chapter 14 - Configuration Management	116
Significant Problems	116
EDIT_PHYSICAL_CONFIGURATION Command May Fail	116
NOS - NOS/VE Dual-State PP Allocation	116
All Controllers OFF Requires Units OFF	117
Notes and Cautions	117
Memory Configurations up to 128 MB	117
Support of 7154 Controller	118
7165-2x/895 or 887-1 Requires a Minimum 4K Page Size	118
NOS/VE Minimum Hardware Requirements in Dual-State Mode	118
State Change of MDI/MTI	119
FCA Levels	119
Chapter 15 - CYBIL	120
Significant Problems	120
INLINE Functions	120
FOR Loop	121
Alignment Inheritance	122
Alignment Ignored for ALLOCATE	122
Small Fields As Reference Parameters	122
Handling Large Constants	123
Chapter 16 - DVS	124
Significant Problems	124
General DVS Problems	124
835 System DVS Problem	125
840 through 860 System DVS Problems	125
990 System DVS Problem	125
Chapter 17 - File Management Utility (FMU)	126
Significant Problems	126
Maximum Block Length for IBM Tape Files	126
Chapter 18 - FORTRAN/VE 1	127
Significant Problems	127
Enforcing Flushing of Terminal Output	127
Manual Errata	128
Ignore References to FORTRAN/VE 2	128
Chapter 19 - HPA/VE	129
Significant Problems	129
HPA/VE Reloads Report's OPEN/CLOSED Status When It Shouldn't	129
HPA/VE Does Not Recognize MAP 5 Engineering Log Error Entries	129
Notes and Cautions	130
Delete Catalogs before Installing NOS/VE 1.2.1 the First Time	130
Running ACCESS_HPA from the System Console	130
Mainframe Errors Logged in NOS or NOS/BE BML	130

Chapter 20 - IM/DM	132
Installation	132
IM/DM Initial Installation	132
IM/DM Upgrade	135
Backup Authority Database	135
Deletion of IM/DM Binaries	135
Initiation of the DM Kernel	135
IM/DM Installation Site-defined Family Name	136
Significant Problems	137
TERMINATE_JOB Command Can Cause Damaged IM/DM Databases	137
Erroneous Diagnostic When FIXED Area Runs Out of Space	138
TERMINATE_JOB command Damages IM/DM Databases	138
Delay in Output of DM Modules	138
Conversion Error on Record Containing Negative Number	139
DISCARD Result Sets Rolls Over at 65771	139
FIND Command in Certain Cases Result in an Error	139
Notes and Cautions	139
File Descriptors May Be Represented by Logical File Names	139
Use PADDING Statement to Create Indexes	140
Use Header Line Information for Submitting PSRs	140
Information Required with PSRs	140
New Version of the FIXC2 Utility for IM/DM	142
Chapter 21 - IM/Quick	143
Significant Problems	143
Unpredictable Results When Output Format Is Incorrect	143
Chapter 22 - LISP	144
Notes and Cautions	144
TERMINATION_ERROR_LEVEL Set to FATAL	144
Chapter 23 - Magnetic Tape Usage	145
Notes and Cautions	145
Unsupported Program Interfaces	145
adding Character for EBCDIC Labelled Tape Files	145
Maximum WORKING_STORAGE_LENGTH for Tape Files	146
Labelled Tape File Block and Record Type Restrictions	146
Maximum Block Length Restrictions for Tape Files	146
Chapter 24 - MAIL/VE	147
Installation	147
MAIL/VE Maintenance	148
MAIL/VE Catalogs and Files	148
Significant Problems	149
Changing Family Names with MAIL/VE	149
Check Mail Fails on WYSE-60	149
The Cursor Is Left in an Unexpected Position After a DELETE	149
Notes and Cautions	149
BACKUP_MAIL and RESTORE_MAIL Procedures	150

Chapter 25 - MALET/VE	151
Significant Problems	151
Buffer Controller Memory Test MY8 Fails	151
MALET/VE Recovery Fails	151
Random Busy Errors Can Occur Running DTI	151
Notes and Cautions	152
MALET/VE Requires a Minimum 4K Page Size	152
Chapter 26 - NAM/VE	153
Manual Errata	153
SYSTEM_PRIVILEGE and CAPABILITY Parameters Have No Effect	153
Chapter 27 - Operations	154
Notes and Cautions	154
VED IJD Operator Display	154
Manual Errata	154
Two New Operator Displays for VEDISPLAY	154
VEDISPLAY DS	155
VEDISPLAY PP	155
Changing Family Names with MAIL/VE	156
Tape Commands	156
CHANGE_DATE (CHAD) Command	157
CHANGE_TIME (CHAT) Command	158
Chapter 28 - PASCAL	159
Significant Problems	159
Nested Variables May Not Be Valid in Outer Procedure/Function	159
Chapter 29 - Permanent File Transfer Facility	160
Significant Problems	160
PTF Requires Family Names of at Least Three Characters	160
Permanent File Transfer Facility Concurrency Problem	160
Error Notification for User Prolog Problems	161
Manual Errata	161
Error NF4 on MANRF or PTF File Access	162
Able to Receive Files But Not Send or Vice Versa	162
Manual Errata	163
Pause Break on PTF Ignored	163
Chapter 30 - Programming Environment	164
Notes and Cautions	164
Creation of C Programs in the Programming Environment	164
Chapter 31 - PROLOG	165
Notes and Cautions	165
Many Predicates Are Now Defined with Arity 2	165
Chapter 32 - Screen Design Facility	166
Notes and Cautions	166
Intensity Attribute	166
Integer and Real Variables	166

Chapter 32 - Screen Design Facility (Continued)	
User Specified Names Must Be SCL Names	166
Chapter 33 - Sort/Merge	167
Significant Problems	167
MINIMUM_RECORD_LENGTH File Attribute	167
Notes and Cautions	167
SMV\$MEMORY_USAGE_LIMIT	167
PAGE_AGING_INTERVAL	168
Chapter 34 - System Command Language (SCL)	169
Significant Problems	169
FORMAT_SCL_PROC Occasionally Makes Lines Too Long	169
FORMAT_SCL_PROC Deletes Some Comments	169
Implicit and Explicit Uses of A File Shouldn't Be Mixed	169
\$PROGRAM Function	170
Constants with Unexpected Values	170
Some Local Files Cannot Be Displayed with DISPLAY_CATALOG	170
Notes and Cautions	170
File Routing Requirements	171
Caution to Be Used with Cycle References	171
REDO - Edit Previously Entered Commands	171
Interactive Prompting (SET_COMMAND_MODE)	171
Manual Errata	172
Sample Prolog Is in Error	173
\$STRREP Can Convert Any Value to a String	173
SET_COMMAND_LIST Has PLACEMENT Parameter	174
Submitting Jobs from NOS	174
Submitting Jobs from NOS/BE	175
SCL Quick Reference Omissions	175
DISPLAY_FUNCTION_INFORMATION Command	176
NOS/VE Welcome Banner	176
\$PROCESSOR Function Produces Different Results	177
Chapter 35 - System Performance and Maintenance	178
Incompatibilities	178
Analyze Dump Restart Files Are Incompatible	178
Significant Problems	178
DISPLAY_BINARY_LOG Cannot Always Access an Active Log	178
Notes and Cautions	179
MAX_WORKING_SET Attribute	179
Online Manual of Site Analyst Examples	179
DISPLAY_SYSTEM_DATA Performance Tool	179
Chapter 36 - Terminal and Screen Management	180
Significant Problems	180
Dual State Transparent Mode Problem	180
NAM Login May Require Additional Carriage Return	180
Errors in Setting NAM/CDCNET Attributes	181
Error Message for Missing TERMINAL_MODEL	181
Caution Regarding Defining Terminals with Field Protect	181

Chapter 36 - Terminal and Screen Management (Continued)	
Incomplete AMP\$PUT_PARTIAL Is Not Flushed	182
Limited Range for Delay Parameters	182
XMODEM Needs Null ATTENTION_CHARACTER for NAM/CDCNET	182
MacConnect File Transfers	182
CDC721 Problems at 19200 BPS Using NAM/VE	183
Disconnect from 2550, Reconnect through CDCNET	183
Notes and Cautions	183
Response After Terminate Break Interrupt	184
PAGE_WIDTH of Zero	184
Terminal Definition of Visual Attributes Restriction	184
Processing of Attributes Relevant to Only One Network	184
Interference between NOS/VE and CDCNET Commands	184
Full-Screen Products Now Support Auto-Tabbing	185
Caution to Users of Zenith Z29 Terminals	186
Terminal Interrupts in Screen-Oriented Products	187
Usage of EDIT_FILE on INTERCOM	188
XMODEM EC Parameter Usage	189
Manual Errata	189
NAM/CCP Terminal Attribute Mapping Error	189
Chapter 37 - Validation	190
Notes and Cautions	190
\$\$SYSTEM Username's Validations and Security	190
Preventing Interactive Access by a User	190
Manual Errata	190
NOS and NOS/VE Password/Validation Relationship Incorrect	191
Order of Epilog Execution Incorrect	191
Appendix A - Release Documentation Bulletins	A-1
Software Release Bulletin (SRB)	A-1
User Impact Bulletin (UIB)	A-1
Software Availability Bulletin (SAB)	A-1
Installation Bulletins (IB)	A-2
Loading Machine Readable Copies	A-2
Appendix B - Site Analyst Documentation Overview	B-1
Appendix C - NOS/BE Dual-State Preparation	C-1
Software Requirements	C-1
Building NOS/VE Dual-State Binaries on NOS/BE	C-1
Step 1 Establish an Installation User Name	C-2
Step 2 Generate NOS/BE Configuration Changes	C-2
Step 3 Install the NOS/VE Release Files	C-3
Step 4 Create the NOS/BE Deadstart File	C-4
Step 5 Redeadstart NOS/BE	C-4
Step 6 Add Information for Partner Jobs	C-4
Step 7 Validate Users on NOS/BE	C-7

Appendix D - Installation Commands	D-1
ACTIVATE_PTF Command	D-2
ACTIVATE_SCF Command	D-3
ACTIVATE_SCFS Command	D-4
DEFINE_PTF Command	D-6
DEFINE_SCF Command	D-7
DEFINE_SCFS Command	D-8
SETUP_INSTALLATION_PROCESS Command	D-9
SETVE Procedure	D-11
SET_VERSION_LEVEL Command	D-14
Appendix E - Critical Display Window Messages	E-1
Unrecovered Disk Errors	E-2
7155-1x Controller	E-2
10395-11 and FA7B4-D Controllers	E-4
7165-2x Controller	E-6
NOS/VE Monitor Detected Errors	E-9
(004) UNCORRECTED IOU ERROR	E-9
(009) UNCORRECTED IOU ERROR	E-9
(004) C170 STATE OS IOU ERROR	E-9
(009) C170 STATE OS IOU ERROR	E-10
(006) FATAL IOU ERROR	E-10
(008) FATAL IOU ERROR	E-10
(105) FATAL CM ERROR	E-10
(20E) FATAL CPU ERROR	E-11
(20F) FATAL C170 STATE UNCORRECTED CPU ERROR	E-11
(210) FATAL C170 STATE EXIT MODE HALT	E-11
(211) FATAL MONITOR DUE	E-11
(212) error	E-12
(213) FATAL MTR MCR	E-12
(216) error	E-12
HR - (217) job mode MCR fault	E-12
(701) ENVIRONMENT WARNING	E-13
(702) POWER WARNING	E-13
(703) POWER WARNING	E-13
(704) POWER/ENVIRONMENT NORMAL	E-14
(705) POWER/ENVIRONMENT NORMAL	E-14
(706) POWER/ENVIRONMENT NORMAL	E-14
TERMINATE NVE COMMAND	E-14
RECOVERY SYSTEM NEEDED	E-14
RECOVERY SYSTEM DONE	E-15
180 terminated by 170 via SCB request	E-15
Dual-state CPU was halted	E-15
error message	E-15
Other Messages	E-17
TASK IS HUNG. HALT OPTION SELECTED.	E-17
n VOLUMES LOW, n VOLUMES OUT date time	E-17
n VOLUMES OUT date time	E-17
PURGED CATALOG path-name DELETED	E-17

Appendix F - Peripheral Reconfiguration and Online Repair	F-1
Capabilities	F-1
Guidelines	F-2
Repair Solution 1	F-3
Repair of 7155 Channel/Controller	F-3
Repair of 7165 Channel/Controller	F-4
Repair Solution 2	F-5
Repair Solution 3	F-13
Repair Solution 4	F-15
Repair Solution 5	F-18
Repair Solution 6	F-19
Repair Solution 7	F-20
Reconfiguration of NOS/VE Deadstart Peripherals	F-21
Appendix G - DISPLAY_SYSTEM_DATA Performance Tool	G-1
Page_Faults Display Option	G-2
Available Queue	G-2
On Disk	G-3
New Pages	G-3
Available Modified Queue	G-3
No Memory	G-3
Low on Memory	G-4
Monitor_Requests Display Option	G-4
Aging_Statistics Display Option	G-5
JM_MM_Statistics Display Option	G-5
Swap_Statistics Display Option	G-7
CPU_Statistics Display Option	G-9
PIO_Statistics Display Option	G-9
Job_Sched_Statistics Display Option	G-11
Appendix H - VED IJD Operator Display	H-1
Appendix I - Common Site Analyst Questions	I-1
Appendix J - NOS/VE REDO Facility	J-1
Activating REDO with BKW/FWD Keys	J-1
Activating REDO without BKW/FWD Keys	J-2
Editing Commands with REDO	J-2
Associating Function Keys with Commands	J-3
Full Duplex Mode	J-3
REDO Command Summary	J-4
Significant Problems	J-4
Appendix K - Online EXAMPLES Manual	K-1
Viewing An Example	K-1
Copying An Example	K-2
Printing An Example	K-4
Accessing An Example by Name	K-5
Trying Out An Example	K-6

NOS/VE 1.2.1 L670 SRB (11/19/86)

Appendix L - NOS/VE Questions Most Frequently Asked L-1

SRB Introduction

This document is the NOS/VE 1.2.1 L670 Software Release Bulletin (SRB). It contains information for installing NOS/VE and its software products. Control Data recommends that the SRB be read in its entirety prior to software installation. You should also ensure that all of your hardware is at the FCA levels indicated in the Configuration Management section of the Software Availability Bulletin (SAB). The release tape ONLINE MANUALS (VSN=NVE007) has a machine readable copy of the SRB. Refer to Appendix A for more information.

1 Audience

This document is written for the site analyst and the person who installs and administrates NOS/VE products. It contains notes and cautions about installation and usage of NOS/VE 1.2.1 L670.

2 Organization

Each chapter of the SRB has information regarding:

1. Special Installation and/or Upgrade Instructions
2. Significant Problems
3. Notes and Cautions
4. Manual Errata
5. Future Considerations

as appropriate. Chapters pertaining only to a site analyst also may have a section on incompatibilities. The first ?? chapters of the SRB describe installation. Following the installation documentation, the SRB is organized alphabetically into chapters on different areas of the operating system and products.

3 Documented But Not Supported Products or Features

The following products or features are documented but not supported for release NOS/VE 1.2.1 L670. Any references to these products or features in the manuals should be disregarded.

1. FORTRAN/VE 2

Chapter 1 Installation Overview

1.1 Significant Problems

The following items describe significant installation problems:

1. Deadstarting After Changing Mode or Memory Parameters
2. Avoid Typeahead at CTI Deadstart Time
3. Upgrading NOS/VE 1.2.1 L670 with Job Recovery
4. SETVE Procedure Fails in Secured Mode
5. Error PF 2044 If All Products Not Installed
6. Inconsistent Results from VERIFY_INSTALLED_SOFTWARE

1.1.1 Deadstarting After Changing Mode or Memory Parameters

The NOS/VE deadstart process in standalone mode expects the content of Central Memory to be as NOS/VE left it from a previous standalone deadstart. If you perform a continuation deadstart of standalone NOS/VE after running dual-state NOS/VE, change memory size or page size, then you must select the CTI option to initialize mainframe memory.

If you do not select the initialize mainframe option, the console will stay blank after CIP execution and deadstart will hang.

PSR NVOI727

1.1.2 Avoid Typeahead at CTI Deadstart Time

NOS/VE standalone deadstart may hang because of typeahead during the CTI portion of deadstart. For this reason, it is recommended to wait for each CTI menu to appear and respond accordingly. The deadstart hang has possibly occurred if the boot screens do not appear in the usual amount of time after the "INITIATING DEADSTART OF NOS/VE" message appears on the screen.

PSRs NVOI726, NVOI765

1.1.3 Upgrading NOS/VE 1.2.1 L670 with Job Recovery

When upgrading a NOS/VE 1.2.1 L670 system to a newer level of NOS/VE 1.2.1 L670, the system will hang trying to load the REQUIRED product tape if active job recovery is enabled and jobs from the previous system are being recovered. This problem can be avoided by terminating all jobs prior to terminating the system before the UPGRADE. The problem may also be avoided by disabling job recovery (via SET_SYSTEM_ATTRIBUTE) on the UPGRADE deadstart.

PSR NVOI641

1.1.4 SETVE Procedure Fails in Secured Mode

The SETVE procedure will not work when executed on a NOS system running in secured mode. A suggested work around would be to run SETVE when NOS is not in a secure mode of operation. Note that when in a secure mode, it is not sufficient to just enter the DSD UNLOCK,username,password. command; deadstarting in an unsecure mode is necessary.

PSR NVOI940

1.1.5 Error PF 2044 If All Products Not Installed

The following error appears in all job logs if optional products like FORTRAN, COBOL, DEBUG, BASIC, and PASCAL are not installed.

```
--ERROR PF 2044-- Subcatalog name ":$SYSTEM.$SYSTEM.xxxxxx  
is unknown.
```

where "xxxxxx" is one of the above mentioned products.

PSR NVOJ516

1.1.6 Inconsistent Results from VERIFY_INSTALLED_SOFTWARE

VERIFY_INSTALLED_SOFTWARE will produce different errors whether it is executed interactively or from a batch job. The errors depend on the products that were installed.

PSRs NVOJ527, NVOJ528, NVOJ529

1.2 Notes and Cautions

The following items describe installation notes and cautions:

1. NOS 2.4.3 Transparent Mode CCP Problem

1.2.1 NOS 2.4.3 Transparent Mode CCP Problem

CCP PSR CC5A471 must be installed for all sites running the dual-state combination of NOS 2.4.3 and NOS/VE 1.2.1 L670. CC5A471 allows the correct operation of transparent mode on NOS/VE 1.2.1 L670. It deletes the page-wait logic from the asynchronous TIP. Most NOS/VE full-screen and graphics applications depend on this change to operator properly.

Code for CC5A471 is available on SOLVER.

1.3 Installation Changes

The following is a brief summary of all of the installation changes that are part of NOS/VE 1.2.1 L670.

1. The documentation of the installation process has changed. In order to keep the release specific installation instructions accurate and documented in only one place, these instructions are now completely contained in this document. The System Performance and Maintenance Manual contains no installation instructions, but does document the Upgrade_Software utility, building of deadstart tapes, and other information that is not specific to installing Level 670.
2. The required release tapes for NOS/VE have been combined to have fewer release tapes.

The following are new tapes:

- | | |
|----------------|---|
| BASE_1 | Contains: the files from the old CDCNET tape, the files from the old REQUIRED tape, the files from the old PFTF tape. |
| BASE_2 | Contains: the files from the old STANDARD tape, the files from the old HARDWARE_MAINTENANCE tape. |
| ONLINE_MANUALS | Contains: the files from both old ONLINE_MANUAL tapes, other online manuals files from the old STANDARD tape. |

The old tapes REQUIRED, STANDARD, PFTF, CDCNET, HARDWARE_MAINTENANCE, ONLINE_MANUALS_I, and ONLINE_MANUALS_II are eliminated.

Besides the elimination of these tapes, this change decreases the installation time and simplifies the process (for example the old CDCNET tape is installed as part of the BASE_1 installation process).

3. One of CDC's installation goals is to completely separate the installation process from the deadstart process. At this release, we have made additional changes to bring us closer in completing this goal. The initial deadstart process no longer contains a dialogue of installation instructions. Most of these instructions were related to installing a NOS/VE network and have now been moved to a new menu driven procedure called DEFINE_NETWORK. This procedure is executed after the deadstart process is completed and is not tied to any installation option.
4. The SET_LOAD_FILES option displayed has been changed and moved. However, the ability to enter commands after the Physical Configuration window has been retained. The system attribute, DEADSTART_INTERVENTION, will cause the deadstart process to pause at this window (see the System Performance and Maintenance Manual for further details). If the operator intervention flag is set to true, a menu of installation option is displayed. The installation options are the same as Level 664, with the addition of a new one called MANUAL. This option, available only on a continuation deadstart, will NOT load the BASE_1 tape and will NOT execute the ACTIVATE_SYSTEM_ENVIRONMENT command.
5. The network activation controls have been simplified into one system attribute, NETWORK_ACTIVATION (see System Performance and Maintenance Manual for details). This means that the operator intervention flag and the dual_state network activator file (\$SYSTEM.DUAL_STATE_NETWORK_ACTIVATOR) no longer reference network activation. The startup commands pertaining to network activation have been moved into a new network startup file to provide greater flexibility. If you are doing an upgrade deadstart you will have to prepare your own network startup file. (See chapter 3 on Upgrading NOS/VE for more details).
6. To decrease the disk space that the installed system occupies, a duplicate copy of the installed products in the installation catalog is no longer kept. This halves the space required to store your system. However, this does mean that to install a corrective tape (BCU), you will be requested by the APPLY_ALL_CORRECTIONS command to mount the release tape for each product that has corrections on the BCU tape. (See chapter 5 on Installing a Correction Package).
7. A new command, SETUP_INSTALLATION_PROCESS, has been created to eliminate many of the setup steps in the Level 664 installation process. This command may change in a future release. It is only used during an initial installation.

Chapter 2

Initial Installation

This chapter describes the initial installation of the NOS/VE operating system. The instructions in this chapter assume that the CYBER Initialization Package (CIP) level 006/L670 has been installed on the device to be used as the system disk. Refer to the CIP User's Handbook for instructions on how to install CIP level 6. This initial installation process assumes that the mainframe that is being installed is not connected to an existing NOS/VE network. For dual-state systems, this chapter also assumes that a compatible version of the NOS or NOS/BE operating system has already been installed.

To complete the installation steps described in this chapter, it is necessary to bring up one terminal to be used as a configuration terminal. For NOS/VE installation in a dual-state environment, the installation of the NOS/VE network is optional, since the configuration terminal can be connected to NAM/CCP or NAM/CDCNET (for NOS) or INTERCOM 5 (for NOS/BE). For a standalone NOS/VE system, the NOS/VE network must be installed. The information provided on the installation of the NOS/VE network is limited to loading the device interfaces (DI's) required to support the configuration terminal and to installing the NOS/VE Timesharing application. Information on configuring printers is not included. For information on how to install the complete NOS/VE network and how to set up printers, refer to the Batch Device User Guide and CDCNET Configuration (**) and Site Administration manual (60461550).

Note that the NOS/VE network installation information in this chapter assumes that your configuration terminal is connected to LIM 0 and PORT 0 of a CDCNET mainframe terminal interface (MTI) or terminal device interface (TDI). If your terminal is not connected to LIM 0 and PORT 0, you must either change the physical connections to LIM 0 and PORT 0 or redefine the DI connection using the Network Operator Utility from the system console. The Network Operator Utility may be used after the CDCNET Device Interface has been loaded. (see Step 7 in this chapter) and is described in the CDCNET Network Operations manual (60461520).

The table on the following page summarizes the installation process described in this chapter.

2.1 NOS/VE Installation Summary

Steps	Purpose	you should
1 through 8	To install the operating system, BASE_1 tape, network products, and configuration terminal.	Go to checklist on the next page and fill in the information appropriate for your site.
9 through 10	To create a new deadstart file that defines the physical configuration and the set of system core commands.	- Read information about PCU on hardware naming conventions, how to define elements, and how to use EDIPC. - Map out your complete physical configuration on paper. - Read information about how to create the deadstart files and deadstart tape in the SPM.
11 through 13	To install the physical configuration, install optional products, and establish the system deadstart disk.	Read the chapter on installing products.
14	To redeadstart to set the boot display option to automatic deadstart.	

2.2 Deadstart Information Checklist

NVExxxx procedures (dual-state systems only)

		Procedure Name
Tape deadstart		_____
Disk deadstart with pause		_____
Disk deadstart without pause		_____
	Deadstart tape	System disk
Channel number	_____	_____
Controller model number	\$ _____	\$ _____
Storage device model number	\$ _____	\$ _____
Equipment number	_____	_____
Unit number	_____	_____
Host connected CDCNET DI		
Element name		_____
Channel number		_____
Serial Number		_____
Element Identifier from list below		\$ _____
MTI on 60Hz power supply - \$2620_210		
MTI on 50Hz power supply - \$2620_211		
MDI on 60Hz power supply - \$2621_210		
MDI on 50Hz power supply - \$2621_211		
System identifier(s) for CDCNET device interface(s)		
MDI		_____
MTI		_____
TDI		_____
CDCNET network identifier		_____
Timesharing titles	_____	_____
System family name for Create_Family		
NOS/VE Family name		_____
NOS/VE Family Administrator		_____

2.3 Step 1 Initiate NOS/VE Deadstart

For dual-state environments (see chapter 6), go to the dual-state console and down the dual-state partner's channels. You may then enter the deadstart procedure that initiates a tape deadstart of NOS/VE. After deadstart is initiated, control of NOS/VE is transferred from the NOS or NOS/BE system console to the NOS/VE console.

For dual-state with NOS, you initiate the deadstart process by typing the following command

n.NVExxxx

where xxxx is the procedure name from the SETVE command.

For dual-state with NOS/BE, type in the following DSD command:

n.X NVE,xxxx,username.

where n is a cleared control point, xxxx is the procedure name from the SETVE command, and username is the ID under which the NVE subsystem runs.

Deadstart of a NOS/VE standalone system is initiated through CIP display screens. The following steps summarize the deadstart initiation process. For more complete information, refer to the CIP User's Handbook.

- A. Set the deadstart panel to select DCF01 as the default DCFILE file. Refer to the hardware reference manual pertaining to your specific mainframe for instructions on how to do this.
- B. Mount the released deadstart tape, VSN001.
- C. For CYBER models 835, 840, 845, 850, 855, and 860, press the button labeled DEADSTART BUTTON on the CC634B console, if the deadstart button option has been installed. If the deadstart button option has not been installed, press the deadstart button on the CC545 console to deadstart.

Please note that on these mainframes, bit 2 of deadstart panel word 12(8) must be set to 1 to select the CC634B as the primary operator console.

For CYBER models 810, 815, 825, 830, 840A, 850A, 860A, and 990, perform the following sequence on the CC634B console:

1. Press the CTRL and G keys simultaneously. The following message will appear:

OPERATOR ACCESS ENABLED

2. Press the CTRL and R keys simultaneously.
3. Press the S key (for the CYBER models 815 and 825 only). The INITIAL OPTIONS CIP screen should appear.

D. When the INITIAL OPTIONS screen appears, enter:

U
(To select the UTILITIES screen)

E. When the UTILITIES screen appears, enter:

I
(To initialize the mainframe)

F. From the INITIAL OPTIONS screen, enter:

O
(To select operator intervention)

The system displays the OPERATOR INTERVENTION screen. The subtitle for the OPERATOR INTERVENTION screen should be NOS/VE DEADSTART. If it is not, enter the O option (TOGGLE OS LOAD STATE) until the correct subtitle appears.

G. From the OPERATOR INTERVENTION screen, enter:

P
(To display the DEADSTART PANEL PARAMETERS screen)

H. From the DEADSTART PANEL PARAMETERS screen, enter:

D=YES
(To select deadstart with operator pause)

Press the backspace key (BS)
(To return to the OPERATOR INTERVENTION screen)

I. From the OPERATOR INTERVENTION screen, enter:

S
(To display the SELECT OS DEADSTART DEVICE screen)

J. From the SELECT OS DEADSTART DEVICE screen, enter:

T
(To select the TAPE option)

K. Some CYBER mainframe models now prompt you to enter the date and time. Enter this information if you are requested to do so.

Enter the time in the 24-hour format: hours.minutes.seconds

NOS/VE 1.2.1 L670 SRB (11/19/86)
Step 1 Initiate NOS/VE Deadstart

12
Initial Installation

09.00.00

Enter the date in the format: month/day/year

12/01/86

The deadstart boot display now appears. You may proceed to define the deadstart tape device according to the instructions in Step 2.

2.4 Step 2 Deadstart Parameters

After you have entered the NVExxxx procedure or selected the CIP display options, the deadstart boot display appears. You use the deadstart boot display to define your default deadstart tape and system disk devices. Information entered in the boot display is stored in the CIP Common Disk Area (CDA) and is saved across deadstarts.

For an installation deadstart, the boot display fields should be defined as follows:

Field	Entry
1	Specify TAPE
2	Specify TRUE
3 through 6	Describe the deadstart tape device
7 through 10	Describe the system disk device

The following shows a sample deadstart boot display:

NOS/VE

Copyright Control Data 1983, 1984, 1985, 1986
Deadstart and System Device Configuration Selections

1. OS Location..... Tape
2. Deadstart pause for operator input... True

Deadstart Device	System Device
3. Channel..... 26(10)	7. Channel 16(10)
4. Controller and \$7021 32	8. Controller and \$7155 12
Storage Device \$679 7	Storage Device \$885 12
5. Equipment Number 0(10)	9. Equipment Number 0(10)
6. Unit Number 0(10)	10. Unit Number 01(10)

NOS/VE Deadstart Command Processor

Enter a menu number to change a value or
Press NEXT to accept parameters and continue the deadstart process

To enter or change a value in the boot display, first type in the number

(from 1 to 10) of the display field. The prompt line at the bottom of the display will describe the options that can be specified for that field. Continue entering display information in this manner until the deadstart and system devices are completely defined, then press the NEXT key to continue with the deadstart. For information on how to define a storage device or on what values can be entered for a particular device, see the Physical Configuration Utility chapter in the System Performance and Maintenance Manual.

2.5 Step 3 System Core Commands

When the system displays the "Enter System Core Commands" prompt, enter the INITIALIZE_DEADSTART_DEVICE (INITDD) and AUTO system core commands as shown in the example dialogue below. The system core commands, including INITDD and AUTO, are described in the System Performance and Maintenance Manual.

INITDD initializes (erases) the system disk device in preparation for installing or reinstalling the operating system. If you choose, you may also define system attributes at this time using the SET_SYSTEM_ATTRIBUTES system core command. The system attributes are described in the NOS/VE System Performance and Maintenance manual. Enter the INITDD and AUTO system core commands:

Enter System Core Commands:

```
inidd vsn001  
auto
```

The system will now proceed to spin the tape and display messages on the console.

2.6 Step 4 Physical Configuration

After a few minutes, the system will ask if you want to define any peripherals in addition to the default physical configuration of one disk drive and one tape drive which was defined on the deadstart boot display.

This is an unconfigured, installation deadstart.
If you answer 'NO' to the question below you will
only have one disk and one tape storage device.
Do you want to define additional peripherals at this time:

Enter yes or no:
yes

If you will be running the NOS/VE network, you must say yes to the prompt in order to define any CDCNET Device Interfaces (DIs) which are connected to your mainframe. We recommend that you delay the definition of any other peripherals until you can use the EDIT_FILE (either through the dual state network or the NOS/VE network) to define the complete physical configuration. To define the CDCNET DI (or any other peripherals), respond with a yes, which will cause execution of the PHYSICAL CONFIGURATION UTILITY (PCU). To alter the physical configuration, you then must enter the EDIT_PHYSICAL_CONFIGURATION (EDIPC) utility to define the DI. Use the ADD_ELEMENT_DEFINITION (ADDED) command to define each peripheral. Below is shown the command to add a CDCNET DI. (PCU and EDIPC are described in the System Performance and Maintenance Manual.) Use the information from the chart at the beginning of this chapter to fill in the correct values.

```
PCU/edipc
PCE/added element_name ei=element_identifier sn=serial_number ..
PCE../          ic=channel_number
```

where "serial number" is the serial number of the equipment.

If additional peripherals are to be defined, do so now. When all definitions are complete, exit EDIPC and install the configuration using the INSTALL_PHYSICAL_CONFIGURATION (INSPC) command of PCU.

```
PCE/quit
PCU/inspc
```

If there are any errors in the configuration, use EDIPC to correct them, then re-install the configuration. When the physical configuration has been installed successfully, enter:

```
PCU/quit
```


2.7 Step 5 Installing the BASE_1 Tape

The system now prompts you to install the BASE_1 operating system tape. Unload the deadstart tape and mount the BASE_1 tape, then enter the ASSIGN_DEVICE command in response to the prompt. The entire dialogue looks like this:

```
      Please assign the BASE_1 tape when prompted.  
      /  
      assign_device $0001 $deadstart_device  
      /
```

When the system has completed reading the tape, the following messages will be displayed:

```
Tape job completed normally
```

```
Initiating UPGRADE_SOFTWARE utility to install files ...
```

The process of installing the BASE_1 files takes a few minutes to complete. During that process, the console displays the names of the files being installed. When finished, the system displays the following messages:

```
BASE_1 product files installed
```

```
Installed prologs and epilogs
```

```
Installed startup commands
```

```
Installed shutdown commands
```

```
Installed network startup commands
```

```
Activating System Environment ...
```

```
Executing startup commands ...
```

```
Startup commands executed
```

```
----- DEADSTART COMPLETE -----
```

2.8 Step 6 Setup Installation Process

The `SETUP_INSTALLATION_PROCESS (SETIP)` command performs many installation initialization steps for you. For complete details of this command, see the Installation Commands Appendix.

To execute this command, enter:

`SETIP family-name user-name`

where "family-name" is the family name and "user-name" is the family administrator user name.

The password is automatically defined for you as the user-name with an "x" added.

2.9 Step 7 Defining the NOS/VE Network

The instructions in this section describe how to install the basic network facilities for your operating system and how to configure a single terminal to use as a configuration terminal for completion of NOS/VE installation. This step is required for standalone systems, but is optional for dual-state users (who may, instead, connect the configuration terminal to NAM/CCP, NAM/CDCNET, or INTERCOM 5).

The NOS/VE network installation described in this section consists of two steps:

- Defining the network configuration.
- Reset the the DI to which the configuration terminal is connected.

The instructions in this step assume that your configuration terminal is connected to LIM 0 and PORT 0 of a CDCNET mainframe terminal interface (MTI) or terminal device interface (TDI). If your terminal is not connected to LIM 0 and PORT 0, you must either change the physical connections to LIM 0 and PORT 0 or redefine the DI connection using the Network Operator Utility from the system console. The Network Operator Utility is available after the initial deadstart is completed (in Step 6) and is described in the CDCNET Operations manual (***).

2.9.1 Defining the Network Configuration

The next four steps are performed using the `DEFINE_NETWORK` commands. Enter the command:

Defn

The following menu of options will be displayed.

Define Network

1. Define the network configuration
2. Define the CDCNET DI configuration
3. Activate the network
4. Define the Timesharing application

Enter a menu selection, QUIT, or ?:

At this point, you may enter a number 1 through 4 as a menu selection, Quit to terminate the `DEFINE_NETWORK` command or a question mark (?) to receive help information about the `DEFINE_NETWORK` command. You should perform the menu options in order.

Each of the menus in DEFINE_NETWORK follow the same guidelines in regards to responding to prompts. They are:

- ⌞ QUIT will terminate the current menu operation and return to the previous menu.
- ⌞ GO (if shown) will cause the parameters to be accepted and an operation will be performed on the parameters, such as configuring a DI, installing the network configuration, etc. Pressing the NEXT key is equivalent to entering GO when GO is displayed in the menu prompt.
- ⌞ ? will cause help information to be displayed.
- ⌞ When you are prompted for a specific parameter value (for example the network identifier or the timesharing title) and GO is not shown in the prompt, you are allowed to press NEXT if you do not want to supply a value.

A brief description of the selections from the DEFINE_NETWORK main menu are give below.

SELECTION 1: Define the network nonfiguration

This selection will prompt you to define the network that exists for each CDCNET DI defined in your physical configuration (step 4). Once this network, which is called a channel network has been defined, the network configuration file is installed.

SELECTION 2: Define the CDCNET DI configuration

This selection will prompt you to supply the system identifier and type (Mainframe Device Interface, Mainframe Terminal Interface, or Terminal Device Interface) for each Device Interface (DI) on the network you are configuring. This will create an initial configuration for each DI which will allow you to log into NOS/VE from the configuration terminal.

SELECTION 3: Activate the network

This selection will activate NAM/VE and its assoicated tasks. No questions are asked.

SELECTION 4: Define the Timesharing application

This option will ask you to supply at least one title which will be used by user's to access the NOS/VE system. The title is specified on the CDCNET CREATE_CONNECTION (CREC) command. You will probably want to define a short and convenient name for your user's to connect to NOS/VE.

2.9.2 Reseting the configuration terminal DI.

You must reset the DI to which your configuration terminal is connected. To reset the DI, open the front panel and reset (pull out and push down) the toggle switch on the Main Processor Board (MPB). The MPB is the first large board from the left side of the cabinet. At this point, the system may require a few minutes to load the DI(s) you have defined. You can check on the progress of the load operation by using the job log console display. To display the job log, enter:

```
ved j1
```

The following error messages will appear during the DI load sequence. These messages do not indicate abnormal conditions and can be ignored:

```
--ERROR NA 172-- File type operator_procedure is unavailable.
```

```
--ERROR NA 11-- DI 80025ssssss reset with reset code 2(16).
```

where "ssssss" is the serial number.

Completion of the DI load is indicated by the following message in the job log:

```
CLOCK SYNC SUCCESSFUL
```

When you see this message, you may proceed to the next step.

(***** add more details about TDI's *****)

2.10 Step 8 Configuration Terminal

For dual-state environments with NOS, the installer logs into NOS/VE through NAM/CCP or NAM/CDCNET follow the normal NOS login process except that VEIAF should be specified for the application name.

For dual-state environments with NOS/BE, the installers logs into NOS/VE through INTERCOM_5 follow the normal NOS/BE login process. When the command prompt appears, type in VEIAF to access the NOS/VE system.

For a standalone or dual-state environment the installers logs into NOS/VE through the NOS/VE network (that is, through NAM/VE), use the following log in procedure:

1. Press the carriage return key a couple times. The following message should appear:

You may enter CDCNET commands

2. Enter a CREATE_CONNECTION command specifying the Timesharing title you defined with the DEFINE_NETWORK command (selection 4) in Step 7 using the:

crec x

where "x" is the timesharing title.

3. Enter the appropriate information in response to the system prompts for your user name, password, and family name, as defined on the SETUP_INSTALLATION_PROCESS command in Step 6:

Connection \$A created.

Enter validation for service access.

User: user1,user1x,family

2.11 Step 9 Define Your Physical_Configuration

In this step, you create the PHYSICAL_CONFIGURATION and PROLOG_FILE files which will be added to the deadstart tape you create in Step 9. System Performance and Maintenance Manual describes the file formats and provides additional information on modifying these files.

The PHYSICAL_CONFIGURATION and PROLOG_FILE files will reside in subcatalog \$SYSTEM.SITE_OS_MAINTENANCE.DEADSTART_COMMANDS.

2.11.1 Create the PHYSICAL_CONFIGURATION file.

1. At the configuration terminal, enter the following commands:

```
setwc $system.site_os_maintenance.deadstart_commands
edif physical_configuration
```

Using the Full Screen Editor (called by the EDIT_FILE command), modify file PHYSICAL_CONFIGURATION to describe your full physical configuration. Specifically, you can:

- ▮ Correct the serial numbers in the element definitions for the system device and the deadstart device.
- ▮ Change the names of the system device and the deadstart device. This step is optional, but you may want to use names other than those that were supplied by the system.
- ▮ Add the DEFINE_ELEMENT subcommands necessary to define the remainder of your physical configuration.

The System Performance and Maintenance Manual explains how to use PCU subcommands to define your physical configuration, and contains examples of configuration definitions. Additional examples are available through the Site Analyst Examples online manual. To access the Site Analyst Examples manual:

- a. From the configuration terminal, enter the following command:

```
explain m=site_analyst_examples
```

- b. On the first menu, select Installation and Configuration Examples.
- c. Select the appropriate example for whatever device you want to configure. After selecting the example, you can view it (using EDIT_FILE) or copy it to another file. (The PRINT option may not work at this time since may not yet have configured a printer.)

2. When you have completed the description of your physical configuration and exited the EDIT_FILE, you should use the PCU to check for errors in your physical configuration definitions. Enter the following commands:

```
pcu
verpc i=physical_configuration
```

If the verification testing finds no errors in your configuration file, enter QUIT to exit the PCU and move on to the next step of creating the PROLOG_FILE. If errors are discovered in the configuration file, you must correct them before continuing. Type QUIT to exit the PCU. Re-enter EDIT_FILE and correct the erroneous definitions. Then re-enter the PCU to check the corrected file using VERIFY_PHYSICAL_CONFIGURATION. Repeat this process until the file is error free.

B. Create the PROLOG_FILE file.

1. Enter the EDIT_FILE.
2. Insert the following line as the first line of file PROLOG_FILE:

```
$local.create_prolog_file file=lcu_mainframe_subcommands
```

3. Begin entering Logical Configuration Utility (LCU) subcommands to initialize the disk volumes in your physical configuration and add them to the system set. For each disk volume in your configuration, enter the following pair of LCU subcommands:

```
inimv element=element_name recorded_vsn='volume_vsn'
addvts member_vsn='volume_vsn'
```

where element_name is the element name of the disk device on which the volume is mounted, and volume_vsn is the 6-digit volume serial number (VSN) you want to assign to this volume. The INIMV (INITIALIZE_MS_VOLUME) and ADDVTS (ADD_VOLUME_TO_SET) subcommands are described in System Performance and Maintenance manual.

Do not include INITIALIZE_MS_VOLUME and ADD_VOLUME_TO_SET subcommands for the system deadstart device. The system deadstart device is initialized and added to the system set by the INITDD system core command.

You may include as many pairs of INITIALIZE_MS_VOLUME and ADD_VOLUME_TO_SET subcommands as required to define all volumes in your configuration. When you are finished describing your disk volumes, terminate the PROLOG_FILE file by typing the following characters in columns 1 and 2:

**

(*** lcu network subcommands ***)

2.12 Step 10 Create a New Deadstart Tape

In this step you use the REPLACE_DS_TAPE_CONFIGURATIONS command to add the files created in Step 9 to the deadstart file. The REPLACE_DS_TAPE_CONFIGURATIONS command searches catalog \$SYSTEM.SITE_OS_MAINTENANCE.DEADSTART_COMMANDS for the physical configuration file and prolog files you created in Step 9. These files are then added to the deadstart tape specified by the INPUT_VSN parameter and the new deadstart file is written to the tape specified by the OUTPUT_VSN parameter. Refer to the System Performance and Maintenance Manual (***) for a complete description of the REPLACE_DS_TAPE_CONFIGURATIONS command.

To create a new deadstart tape:

1. Mount the unconfigured deadstart tape (NVE001).
2. From the configuration terminal, enter the following commands to add the modified and converted files to your deadstart tape:

```
setcl a=$system.osf$builtin_library
reptdc ivsn=vsni ovsni=vsni density=keyword
```

IVSN specifies the VSN of the input tape; in this case, the unconfigured deadstart tape. OVSNI specifies the VSN of the output tape. If you omit OVSNI, the input tape is overwritten; however, it is recommended that you use a new output tape.

The DENSITY parameter is applicable to the output tape only. Keyword values that can be specified for the DENSITY parameter are:

MT9\$800	9-track, 800 cpi
MT9\$1600	9-track, 1600 cpi
MT9\$6250	9-track, 6250 cpi

The REPLACE_DS_TAPE_CONFIGURATIONS command searches catalog \$SYSTEM.SITE_OS_MAINTENANCE.DEADSTART_COMMANDS for files PHYSICAL_CONFIGURATION and PROLOG_FILE and adds them to the deadstart tape specified by OUTPUT_VSN.

3. Mount the new deadstart tape (specified by OUTPUT_VSN) with a write ring when the system prompts you to do so.
4. From the terminal, log out of the system by entering:

```
logout
```

5. From the system console, enter the following command to turn off the file and catalog permissions created in Step 6:

chate configuration_file_access=off family-name=name user-name=name

where "FAMILY-NAME" and "USER-NAME" specify the names defined in Step 6.

2.13 Step 11 Redeadstart the System

Redeadstarting the system from the tape created in Step 10 allows the PCU and LCU commands to be entered from file PROLOG_FILE. In other words, at the end of this deadstart your full physical configuration (as described by the PCU and LCU commands you specified in Step 9) will be activated.

From the system console:

- A. Terminate the system by entering the `TERMINATE_SYSTEM` command:

`terminate_system`

- B. Initiate a deadstart of NOS/VE.

For dual-state systems, go to the NOS or NOS/BE console and enter the deadstart procedure that specifies a tape deadstart, then go to step 11C.

For standalone systems, initiate CIP execution by performing the following steps:

1. At the console, enter CTRL G followed by CTRL R
2. When the INITIAL OPTIONS screen appears, enter:

0
(To select operator intervention)

The system displays the OPERATOR INTERVENTION screen. The subtitle for the OPERATOR INTERVENTION screen should be NOS/VE DEADSTART. If it is not, enter the 0 option (TOGGLE OS LOAD STATE) until the correct subtitle appears.

3. From the OPERATOR INTERVENTION screen, enter:

P
(To display the DEADSTART PANEL PARAMETERS screen)

4. From the DEADSTART PANEL PARAMETERS screen, enter:

D=YES
(To select deadstart with operator pause)
back space (BS) (To return to the OPERATOR INTERVENTION screen)

5. From the OPERATOR INTERVENTION screen, enter:

S
(To display the SELECT OS DEADSTART DEVICE screen)

6. From the SELECT OS DEADSTART DEVICE screen, enter:

T
(To select the TAPE option)

C. When the deadstart boot display appears, it should display the same values you entered in Step 2. If you are deadstarting from the same tape device, press NEXT to accept the displayed values. If you are deadstarting from a different device, make the appropriate changes to the device description, then press NEXT.

D. When the "Enter System Core Commands" prompt appears, enter:

```
useic false
go
```

After you enter the GO system core command, the system displays the PCU prompt, which asks if you want to modify the physical configuration just installed. Respond by entering NO:

```
Do you want to change the physical configuration?
Enter yes or no
no
```

E. The final item that must be done is to install the LCU_MAINFRAME_SUBCOMMANDS file created in Step 9. To do this, respond as follows when the LCU prompt appears:

```
Do you want to change the logical configuration?
Enter yes or no
yes
```

```
LCU/ include_file f=$local.lcu_mainframe_subcommands
LCU/ quit
```

After you exit the Logical Configuration Utility, wait for the following message to signal that deadstart is complete.

----- DEADSTART COMPLETE -----

2.14 Step 12 Install the Products

At this point, you install the product tapes for the remaining required and optional products by executing Upgrade Software (UPGS) at the system console. An `INSTALL_PRODUCT` subcommand must be entered for each tape to be installed. UPGS is described in System Performance and Maintenance Manual.

The installation of CDCNET, IM/DM, MAIL/VE, and the Accounting Analysis System require additional installation steps. Refer to chapter 3 for information on how to install these products.

The following example installs the FORTRAN and COBOL products:

```
upgs
  insp fortran
  insp cobol
quit
```

The system prompts you to load the tape for each product specified in an `INSTALL_PRODUCT` subcommand. When it does, enter an appropriate `ASSIGN_DEVICE` command and load the product tape. The product tapes and their associated VSN's are listed in chapter 4.

After installing all of the products, you may wish to execute the UPGS subcommand, `Display_Installed_Software`, to list all of the tapes that were installed. For details on this command, refer to the System Performance and Maintenance manual.

```
disis
```

2.15 Step 13 Copy the Deadstart File to Disk

This step copies the deadstart file to the system disk for use in continuation deadstarts.

A. Mount the deadstart tape.

B. Enter the ESTABLISH_DISK_BASED_SYSTEM command at the system console:

```
estdb
```

C. Assign the deadstart tape to the job by entering an appropriate ASSIGN_DEVICE command, as shown in the following example dialogue:
(during Step 9, we changed \$DEADSTART_DEVICE to j50)

```
assd $0005 j50
```

When the tape has been read and unloaded, the following messages are displayed:

```
Establishing disk based system ...
```

```
Please wait for completion message.
```

The following message indicates that the deadstart file has been successfully established on disk:

```
Disk based system has been established
```

Note that it may take two or three minutes for this message to appear.

2.16 Step 14 Automatic Deadstart

This is an optional step that sets the deadstart boot display to specify an automatic deadstart (that is, a deadstart executed from the default disk device established in Step 13).

- A. Enter the `TERMINATE_SYSTEM` command to terminate the system:

`terminate_system`

- B. Initiate a NOS/VE deadstart.

For dual-state systems, go to the NOS or NOS/BE console and enter the `NVExxxx` procedure that specifies a disk deadstart with operator pause.

For standalone users, initiate CIP execution by performing the following steps:

1. At the system console, enter CTRL G, followed by CTRL R.
2. When the INITIAL OPTIONS screen appears, enter:

0
(To select operator intervention)

The system displays the OPERATOR INTERVENTION screen. The subtitle for the OPERATOR INTERVENTION screen should be NOS/VE DEADSTART. If it is not, enter the 0 option (TOGGLE OS LOAD STATE) until the correct subtitle appears.

3. From the OPERATOR INTERVENTION screen, enter:

P
(To display the DEADSTART PANEL PARAMETERS screen)

4. From the DEADSTART PANEL PARAMETERS screen, enter:

D=YES
(To select deadstart with operator pause)

Press the backspace key (BS)
(To return to the OPERATOR INTERVENTION screen)

5. From the OPERATOR INTERVENTION screen, enter:

S
(To display the SELECT OS DEADSTART DEVICE screen)

6. From the SELECT OS DEADSTART DEVICE screen, enter:

T
(To select the TAPE option)

- C. When the deadstart boot display appears, it should display the same values you entered in step 10. Change the values as follows:

Field	Entry
1	Specify DISK
2	Specify FALSE
7 through 10	Describe the system disk device

When you have correctly entered the values on the boot display, press the NEXT key to continue with the deadstart. When the following message appears, the deadstart is complete:

----- DEADSTART COMPLETE -----

The system is now set up to perform an automatic deadstart the next time you need to deadstart the system.

For dual-state systems, this means you can deadstart NOS/VE by entering the deadstart procedure that specifies a disk deadstart without operator pause.

For standalone systems, this means you can deadstart NOS/VE by entering the A (Automatic) option on the CIP INITIAL OPTIONS screen.

2.17 After You Have Installed NOS/VE

When you have completed the NOS/VE installation, there are some additional tasks you may want to perform.

- Change the password for family :\$SYSTEM user \$SYSTEM, using the Family Administration Utility (describe in the Family Administration manual). It is recommended that the default password be changed immediately after the installation is complete.
- Create additional user validations. Use the Family Administration Utility (described in the Family Administration manual) to create additional user validations for the system.
- Change the password on the family administrator created in Step 6.
- Complete the installation and configuration of NAM/VE. Chapter 4 provides information on installing the Status and Control Facility (SCF), the Status and Control Facility Server (SCFS), and PTF. SCF and SCFS must be installed to allow the printing of files in a NOS/VE standalone system.
- Complete the CDCNET configuration. First, go to chapter 3 and read the information pertaining to CDCNET. Then go the CDCNET Configuration and Site Administration manual for further installation instructions.
- Optionally perform site tailoring, as described in the System Performance and Maintenance manual.
- Do a permanent file back up of your system using the Permanent File Backup Utility (described in the NOS/VE Operations).

Chapter 3

Upgrade Installation of NOS/VE

This chapter describes upgrading NOS/VE 1.2.1 Level 664 to NOS/VE 1.2.1 Level 670. If you are using NOS/VE Network, this upgrade requires you to add the CDCNET Device Interface to the physical configuration and alter the Network configuration commands between host and device interface. This upgrade does requires adding CDCNET Device Interface to the physical configuration if you are using NOS/VE Network.

For dual-state systems, this chapter assumes that a compatible version of the NOS or NOS/BE operating system has already been installed. There is no difference, at Level 670, in the steps required to upgrade a NOS/VE system or a NOS/VE system with a dual-state partner. It is also suggested that timesharing access to NOS/VE from a dual-state partner be disabled during the upgrade process.

3.1 Step 1 Miscellaneous Changes

There are several steps which must be performed to prepare for the upgrade to Level 670. It is suggested that these steps be performed on Level 664 before deadstarting NOS/VE Level 670.

A. Build a configured deadstart tape.

- ▮ Move current configurations to the 670 deadstart tape.
- ▮ If running the NOS/VE network, add `define_element` commands for the DI's to configuration prologs for mainframes on which you run a NOS/VE Network.
- ▮ Change LCU Network commands

B. Change NOS/VE Permanent Files

The start up commands that are executed at Level 664 are in the file `$SYSTEM.START_UP_COMMANDS`. This file contains commands for the NOS/VE network as well as for other system operations. For Level 670, a new file called `$SYSTEM.NETWORK_STARTUP_COMMANDS`, must be created. The network startup commands file is to contain only commands that should be executed when the NOS/VE network is activated. These network commands are remove from the `$SYSTEM.START_UP_COMMANDS` file.

- ▮ Move any network commands that you have changed into the new file `$SYSTEM.NETWORK_STARTUP_COMMANDS`. The `NETWORK_STARTUP_COMMANDS` file will have to be created. The `SCL IF` statement testing (found in the old `START_UP_COMMANDS` file) for `$NAMVE_ACTIVE` should not be included. This test is performed elsewhere and is no longer needed here.
- ▮ Remove all network commands from the file `$SYSTEM.START_UP_COMMANDS` and the `IF` statement block.
- ▮ The `DUAL_STATE_NETWORK_ACTIVATOR` file is no longer used. You can delete it from your system. The new system attribute `NETWORK_ACTIVATION` performs many of the same functions.

3.2 Step 2 Deadstart NOS/VE Level 670

Using the newly built configured NOS/VE 670 deadstart tape, deadstart your system with the operator intervention flag set. Remember to select the tape deadstart option at the CIP display.

On a dual_state system you should first disable the VEIAF and IRHF access and then deadstart the dual_state partner by

(more details to follow)
(from operations handbook, 11-2 and 11-3).

3.3 Step 3 System Core

When the system displays the "Enter System Core Commands" prompt, enter the new NETWORK ACTIVATION system attribute command to disable network activation followed by the GO command. This will keep the operator intervention flag set to true. The network must not be active during the upgrade process.

```
Enter System Core Commands:
setsa network_activation 0
useic false
go
```

After you enter the GO command, the system displays the PCU prompt, which asks if you want to modify the physical configuration. Respond by entering NO.

```
Do you want to change the physical configuration?
Enter yes or no
no
```

After you enter QUIT, the system displays the LCU prompt, which asks if you want to modify the logical configuration. Respond by entering NO.

```
Do you want to change the logical configuration?
Enter yes or no
no
```

3.4 Step 4 Installation Option Menu

The installation selection menu is then displayed. As this is an UPGRADE installation, you should choose option 2.

A NORMAL installation was selected.

Available installation options are:

1. NORMAL: No further installation processing, the system environment is activated.
2. UPGRADE: The BASE_1 product tape is loaded and the system environment is activated.
3. INSTALL
DEFERRED FILES: Install the files previously deferred in an earlier deadstart.
4. MANUAL: No further installation processing, the system environment is NOT activated.

Enter the option number to change the selected installation type, or

Press NEXT to perform the selected installation:

2

After this menu the following prompt will appear. You should answer this prompt with a NO if you do not wish to save the previously installed version (i.e. Level 664) of the products. If you do choose to save the old versions of the products the disk space will be increase.

Do you want to retain previous versions of the software to be installed?

/

no

3.5 Step 5 Installing the BASE_1 Tape

The system now prompts you to install the BASE_1 operating system tape. Unload the deadstart tape and mount the BASE_1 tape, then enter the ASSIGN_DEVICE command in response to the prompt.

The process of installing the BASE_1 files takes a few minutes to complete. During that process, the console displays the names of the files being installed. When finished, the system displays the following messages:

BASE_1 product files installed

Activating System Environement ...

Executing startup commands ...

Startup commands executed

----- DEADSTART COMPLETE -----

If running a dual-state partner, you should disable the system tasks for RHINPUT, RHFOUTPUT, and IAFEX.

(** details to follow **)

3.6 Step 6 Installing the Products

This step is identical to Step 12 of the initial installation of NOS/VE. Please refer to this step in chapter 2 for additional assistance.

If you are using the NOS/VE network, enter the Activate network command as shown below:

```
Activate_network true
```

Many messages will be displayed. The key messages to watch for are displayed below.

```
NAM/VE activated  
Executing network startup commands ...  
Network startup commands executed
```

If you are using a network from the 170 dual_state partner, return to the 170 console and enable interactive and batch access as shown below:

```
( *** copy from pages 11-2 and 11-3 *** )
```

```
(details to follow)
```

3.7 After You Have Upgraded NOS/VE

When you have completed the NOS/VE upgrade, there are some additional task you may want to perform.

- ▮ Copy the upgraded system to your system disk. (use the Establish_disk_base system command).

Chapter 4 NOS/VE Product Installation

This chapter describes the installation of the NOS/VE products that are network related or do not require special installation instructions. Products that do require special installation instructions are described in later chapters of this document.

4.1 Standard and Optional Product Installation

Various products can be installed by your site without performing an installation deadstart. You can install the products by executing the Upgrade Software Utility at the system console and entering the `INSTALL_PRODUCT` subcommand for each product that you wish to install. As the `INSTALL_PRODUCT` subcommand generates NOS/VE tape requests, be sure that a tape drive is available to NOS/VE for completion of the product installation. Additional information regarding `Upgrade_Software` and the `INSTALL_PRODUCT` subcommand appears in the System Performance and Maintenance Manual.

The following table lists the names of the products currently available for NOS/VE, the corresponding volume serial number (VSN) for each product tape, and an indication of whether the product is required, standard, or optional. Required products must be installed. Standard products are automatically supplied with the system, although you must explicitly install them. Optional products are not supplied as part of the standard system and must be purchased separately by your site.

List of NOS/VE Products

Tape VSN	Product Name	Product Type
NVE001	NOSVE DEADSTART	required
NVE003	BASE_1	required
NVE004	BASE_2	required
NVE007	ONLINE MANUALS	standard
NVE009	FORTRAN	optional
NVE010	COBOL	optional
NVE011	APL	optional
NVE012	CYBIL	optional
NVE014	LISP	optional
NVE015	MANUALS_MAINT_I	standard
NVE016	MANUALS_MAINT_II	standard
NVE018	MAPV	optional
NVE020	PROLOG	optional
NVE022	ENVIRONMENTS	optional
NVE023	ACCOUNTING_UTILS	optional
NVE024	BASIC	optional
NVE025	PASCAL	optional
NVE026	QUICK	optional
NVE027	IM_DM_BASE_I	optional
NVE028	IM_DM_BASE_II	optional
NVE054	IM_DM_BASE_III	optional
NVE029	IM_DM_QUERY	optional
NVE030	IM_DM_REPORT	optional
NVE031	IM_DM_PROCESSOR	optional
NVE032	IM_DM_DLI	optional
NVE039	ACCT_ANALYSIS_SYS	optional
NVE055	LIB99	optional
NVE056	SCREEN_DESIGN_FACILITY	optional
NVE041	PTF_QTF	optional

4.2 Network Installation

Installation of the NOS/VE network is required for standalone configurations and is optional for dual-state configurations. If you have a dual-state configuration and do not install the NOS/VE network during initial installation of the operating system, you may do so at a later time.

The system files required to install the NOS/VE network are already on the system. To define the network, just enter the `DEFINE_NETWORK` command at the system console. `DEFINE_NETWORK` defines CDCNET, NAM/VE, and the Timesharing application. Please refer to Chapter 2, Step 7 for additional details.

4.3 Installation of PTF, SCF, and SCFS

After you install the NOS/VE network, there are three other network applications you may install. These are:

- the Status and Control Facility (SCF)
- the Status and Control Facility Server (SCFS)
- the Permanent File Transfer Facility (PTF)

Use the following Upgrade_Software subcommands to define these applications:

```
DEFINE_PTF
DEFINE_SCF
DEFINE_SCFS
```

The definition of PTF, SCF, and SCFS are maintained across all deadstarts except initial installation.

Use the following commands to activate these applications:

```
ACTIVATE_PTF
ACTIVATE_SCF
ACTIVATE_SCFS
```

Activation of these applications is not maintained across deadstarts; you must reactivate each application after every deadstart if you want to use these products. For this reason, it is recommended that you place the ACTIVATE_PTF, ACTIVATE_SCF, and ACTIVATE_SCFS commands in the \$SYSTEM.NETWORK_STARTUP_COMMANDS file described in the System Performance and Maintenance Manual.

The following subsections describe the steps to use in defining and activating PTF, SCF, and SCFS. The definition and activation commands are described in detail following these subsections.

4.4 SCF and SCFS Installation

The SCF and SCFS applications provide the batch device support required for printing files in a CDCNET Network. SCF must be defined and activated on each mainframe. SCFS is required on only one mainframe in a network; however, it may be desirable to define it on every mainframe.

If SCFS is defined on more than one mainframe in a network, unique names must be used for the control facility names. Unique names must be used for application names and system task names define within the same mainframe.

This means that the default name for the SCFS application can be used only once in a given network.

To deactivate or reactivate the SCF and SCFS system tasks, use the DEACTIVATE_SYSTEM_TASK or the ACTIVATE_SYSTEM_TASK commands, supplying the appropriate task name:

```
deactivate_system_task taskname
activate_system_task taskname
```

where "taskname" is the task you wish to activate or deactivate.

Perform the following steps to define and activate SCF and SCFS:

1. Define SCF and SCFS.

If you intend to install SCFS on only one mainframe in a network, you may use the default value for the APPLICATION_NAME parameter of the DEFINE_SCFS command. If you are defining more than one copy of SCFS in a network, each copy of the application must have a unique application name; therefore, the default name may be used only once. Enter the following sequence of commands to define SCF and SCFS:

```
upgs
  define_scf
  define_scfs application_name
quit
```

where "application_name" is the name defined for the SCFS server application.

The default task name created by the DEFINE_SCF command is SCF_CLIENT. The default task name created by the DEFINE_SCFS command is SCF_SERVER. You must know these task names when performing any subsequent deactivations or reactivations of these tasks.

2. Activate SCF and SCFS by entering

```
activate_scf
activate_scfs application_name
```

where "application_name" is the same name used on the DEFINE_SCFS command in Step 1.

The ACTIVATE_SCF and ACTIVATE_SCFS commands should also be placed in the network startup commands file for automatic activation across deadstarts (see Site Tailoring in the System Performance and Maintenance Manual).

4.5 PTF Installation

PTF provides a means of transferring files between mainframes on a network. Before you can install PTF, you must ensure that each family to be accessed through PTF has a unique family name in the network.

Use the following sequence of steps to install PTF. These steps must be performed on each mainframe to be accessed through PTF.

1. Install and define the PTF application. Mount the PTF/QTF product tape and enter the following Upgrade Software Utility commands:

```
upgs
  insp ptf_qtf
  define_ptf (list-of-family-names)
quit
```

where the "list-of-family-names" lists every family on this mainframe to be accessed by PTF.

As a result of the INSTALL_PRODUCT subcommand, the system prompts you to assign the PTF/QTF product tape. You respond by entering an appropriate ASSIGN_DEVICE command. The system then continues processing the DEFINE_PTF command. On completion, the system displays the following message:

PTF application is installed

2. Create the FILE_TRANSFER job class.

The PTF server executes in this job class, and users must be validated for this job class in order to use PTF. Users must also be validated to use the EXPLICIT_REMOTE_FILE and IMPLICIT_REMOTE_FILE capabilities (refer to the Family Administration manual for information on these capabilities).

To create job class FILE_TRANSFER, enter the command:

```
crejc i=9 n=file_transfer dv=batch maxij=40 ..
maxaj=40 iic=true
```

The index parameter must be set to an available job index. You can also tailor any of the parameters to meet site requirements. For example, you may want to raise the job priority. Refer to the NOS/VE System Performance and Maintenance manual for more information on these parameters. Because the CREATE_JOB_CLASS command must be entered after every deadstart, it is recommended that you place the command in the network start up commands file described in System Performance and Maintenance Manual.

3. Define and activate the PTF server system task.

Enter:

activate_ptf

The ACTIVATE_PTF Command should also be placed in the network startup commands file for automatic activation across deadstarts.

The name of the system task just defined and activated is OSA\$FILE_TRANSFER_SERVER. You will need to know this name for any subsequent deactivations and reactivations of the PTF server system task.

Chapter 5

Installing a Correction Package

This chapter describes the steps you must follow to install a correction package on your system. A correction package, also referred to as a BCU or Batch Critical Update, contains corrections to all NOS/VE products that CDC has released corrective code for. Each BCU contains all previous BCUs for a system release level.

The `UPGRADE_SOFTWARE` (UPGS) subcommand, `APPLY_ALL_CORRECTIONS`, will automatically correct all products that you have installed on your system that have correction code on the BCU tape. This process requires each product that is to be corrected be reloaded into the installation catalog. After the corrections for a product have been applied, that product will be deleted from the installation catalog. This entire process is handled automatically by `APPLY_ALL_CORRECTIONS`.

If you wish to install only some of the corrections from the BCU, you may use the UPGS subcommand `APPLY_VIRTUAL_STATE_CORRECTION` along with other subcommands (see the System Performance and Maintenance Manual (**)) for additional details) to aid you in completing this task.

Correction packages may be installed from tape or from disk. To install correction packages from tape, complete all the steps described in this chapter. To install packages from disk, perform all steps except 2 and 3.

If the correction package contains changes to any of the operating system files, a deadstart file (tape or disk) is automatically created. If this deadstart file is a disk file it must be copied to a tape before it can be used to deadstart the system.

At least one tape drive is necessary to complete a BCU installation.

Instructions for installing corrections to NOS dual-state code (NOS binaries) are found in the NOS Installation Handbook. For instructions on installing NOS/BE corrections, NOS/BE Installation Handbook.

Installing a Correction Package

5.1 Correction Package Installation Process

Follow the steps below to install the correction package. Before you begin, be sure you have a tape drive available to NOS/VE.

1. From the system console, enter the Upgrade_Software (UPGS) by typing:

```
upgrade_software
```

2. Mount the tape that includes the correction package. (If the correction package is on disk, go to Step 4.)
3. Convert the tape file to a disk file by entering:

```
rescp vsn=xxxxxx ..  
corp=$system.correction-package
```

where "vsn" and "correction-package" parameters are required. The CORRECTION_PACKAGE parameter can indicate any local or permanent file that contains the correction package.

4. Apply the corrections by entering the APPLY_ALL_CORRECTIONS subcommand. Type in:

```
appac correction-package=$system.correction_package ..  
new-ve-deadstart-tape='xxxxxx'
```

where "correction-package" is the local or permanent file that contains the correction package and "new-ve-deadstart-tape" is the vsn for a new deadstart tape. NOTE a deadstart tape is only created when a correction package contains corrections to operating system files. If this parameter is not specified and changes exist a deadstart file is written instead of the tape.

You will be prompted to mount the release tapes for any products that you have installed that have corrections on the BCU. Be sure you are assigning the correct release tape to each prompt by checking the vsn identifier on the tape with the vsn specified in the prompt.

You may wish to monitor the batch job with the VED AJ command. The batch job will complete after all the corrections have been applied or an error has been detected. If the job does abort, you will be notified via an operator request. The lack of this request usually means the job has ended successfully. In either case, to get an understanding of exactly what occurred, (check the UPGRADE LOG or the \$JOB_OUTPUT_QUEUE).

5. Exit UPGS typing:

quit

6. Terminate the system by typing:

terminate_system

7. Deadstart the system with the operator pause set to TRUE. To specify operator pause for standalone deadstarts, specify P=YES on the DEADSTART PANEL PARAMETERS CIP display. For dual-state, use a deadstart procedure that initiates deadstart with operator pause.

8. When the "Enter System Core Commands" prompt appears, type in:

go

9. When the installation options display appears (you may ignore the PCU and LCU displays), select option 3, the install deferred files option.

After the following message appears, the system is ready for production.

----- DEADSTART COMPLETE -----

If a new deadstart tape was created in Step 4, refer to System Performance and Maintenance Manual for instructions on creating a disk-based system from a deadstart tape.

For further information on the UPGS, refer to the System Performance and Maintenance Manual.

Chapter 6

Dual-State Preparation

This chapter is applicable for sites performing a first time installation of NOS/VE in a dual-state environment (NOS or NOS/BE). Installation of NOS/VE on a dual-state system requires that the dual state operating system already be installed and running. NOS installation is documented in the NOS Installation Handbook and NOS/BE installation is described in the appendix of this document.

The following installation steps create and execute the dual state procedures that perform bootstrap functions for the NOS/VE installation deadstart. Once these steps have been performed, go to chapter 1, NOS/VE Installation, and proceed with the NOS/VE installation process described in that chapter.

6.1 Step 1 Generate NOS/VE Deadstart Procedure Files

You generate NOS/VE deadstart procedure files using the dual-state procedure SETVE (the description of this command is at end of this chapter). On NOS, the SETVE creates a procedure as an indirect access file under the NOS user name SYSTEMX. Whereas, on NOS/BE the SETVE procedure creates a procedure file as a NOS/BE permanent file with the ID specified by the UN parameter.

Deadstart procedure files created by SETVE are executed from the dual-state system console. Their purpose is to initiate the NOS/VE deadstart process. You can create more than one deadstart procedure file, each specifying a different set of deadstart options.

Parameters on the SETVE procedure allow you to define whether or not the deadstart process will pause for operator input and which set of system core commands will be used. For installing and upgrading the system, you must have a deadstart procedure that pauses for operator input. The pause is optional for a continuation deadstart. Create as many deadstart procedures as you need. At a minimum, you will want three deadstart procedures for each configuration you have on the deadstart input file:

1. A deadstart from tape
2. A deadstart from disk with operator pause
3. A deadstart from disk without operator pause

6.2 Step 2 Reserve VE Elements (NOS only)

You can automatically reserve the NOS/VE equipment and channels through the appropriate EQPDECK and IPRDECK entries. It is still recommended that you do the following to determine if the NOS-defined equipment to be used by NOS/VE has been downed.

1. Request the DSD equipment display by entering the DSD command:

E,A.

2. All disk and tape units appearing in this display that are to be used by NOS/VE must have a status of DOWN. If the disk or tape units for NOS/VE do not have a status of DOWN, enter the DSD DOWN command to make them available for NOS/VE. The format of the DSD command is:

DOWN,EQ=est.

where "est" is the octal equipment status table (EST) number of the equipment. (EST numbers appear in the first column of the display.)

Do the following to determine if all channels that are to be used by NOS/VE are available.

1. Request the DSD channel status display by entering the DSD command:

W,C.

2. All channels that are to be used by NOS/VE must have a status of DOWN. If any channels need to be reserved for NOS/VE, use the DSD DOWN command to do so. The format of the DSD command is:

DOWN,CH=ch.

where "ch" is the octal channel number. Enter this command for each channel to be used by NOS/VE.

6.3 Step 3 Enable the NVE Subsystem (NOS only)

If you did not enable the NVE subsystem using an IPRDECK ENABLE command, you must do so by entering the following DSD operator command:

```
SUBSYST.  
L.ENABLE,NVE,xx.  
L.END
```

where "xx" represents the control point number assigned to NVE.

This entry can also be used to change the control point assigned to NVE if NVE was enabled by an IPRDECK ENABLE command.

To proceed with the installation of NOS/VE, turn to chapter 1 and follow the installation steps described in that chapter.

Chapter 7 Accounting

7.1 Installation

After you have installed the Accounting Analysis System product tape, the system administrator must run the `INSTALL_ACCOUNTING_ANALYSIS` procedure to complete the installation of the system. This procedure starts the Accounting Analysis System and should be completed before any jobs are allowed to run. The `INSTALL_ACCOUNTING_ANALYSIS` procedure is documented in the NOS/VE Accounting Analysis Manual (60463923).

7.2 Notes and Cautions

The following items describe accounting notes and cautions:

1. Ignore ERROR Issued by `INSTALL_ACCOUNTING_ANALYSIS`

7.2.1 Ignore ERROR Issued by `INSTALL_ACCOUNTING_ANALYSIS`

When installing the accounting analysis system via the `INSTALL_ACCOUNTING_ANALYSIS` command, the system displays the following error:

```
--ERROR-- File unique-file-name : AMP$START not implemented.
```

where "unique-file-name" is a string representing a unique file name. This error message can be safely ignored. The accounting analysis system will be correctly installed.

7.3 Manual Errata

The following items are Manual Errata in the area of Accounting:

1. Counters Raised from 63 to 256

7.3.1 Counters Raised from 63 to 256

The allowable value for "n" in the counter descriptions is now 1 through 256 rather than 1 through 63. This is documented on pages 3-5, 3-11, 3-16, and 4-16 in the NOS/VE Accounting Analysis Manual (60463923).

Chapter 8

Advanced Access Methods (AAM)

8.1 Significant Problems

The following items are significant problems in the area of AAM:

1. Sharing Files on Dual Processor Systems
2. Certain File Attributes Not Supported

8.1.1 Sharing Files on Dual Processor Systems

On dual processor systems, sharing of INDEXED_SEQUENTIAL or DIRECT_ACCESS files across asynchronous tasks can cause problems if the job has chosen

```
set_multiprocessing_options mp=on
```

If such files are to be shared in a dual processor system, the use of jobs rather than tasks to accomplish this is recommended. Sharing across asynchronous tasks (i.e. tasks started with WAIT=FALSE) works only if multiprocessing in the job is avoided.

PSR AA8A550

8.1.2 Certain File Attributes Not Supported

The combined file attributes EK = TRUE and KP <> 0 are not supported as part of the compression feature with NOS/VE 1.2.1 L670. Attempting to specify this combination of file_attributes, will result in the following diagnostic message:

The combination of CPN=YES, EK=TRUE, and KP <> 0 for compression feature is not supported at this level.

PSR AA8A736

Chapter 9 APL

9.1 Notes and Cautions

The following items are notes and cautions in the area of APL:

1. END_PARTIAL_CHARACTER Attribute Reset by APL
2. Conversion Utility

9.1.1 END_PARTIAL_CHARACTER Attribute Reset by APL

APL changes the terminal attribute END_PARTIAL_CHARACTER to NUL. APL does not restore END_PARTIAL_CHARACTER to its previous value when it exits. If the user wishes it set to line feed after exiting from APL, this can be done with the CDCNET command:

```
<ncc>CHANGE_TERMINAL_ATTRIBUTES EPC=LF
```

where <ncc> is the Network Command Character.

9.1.2 Conversion Utility

APL/VE provides a utility (SCL command CONVERT_APL2 or workspace .APL.CONVERT_APL2) for converting CYBER 170 or 180 APL2 workspaces and APL2 structured files. For users running on a dual-state system, this utility provides the option of getting workspaces and files from NOS. In a standalone system, the utility requires any APL2 workspaces or files that you wish to convert, to exist as NOS/VE files which must have been brought over from NOS via the GET_FILE command with DC=B60.

If you have installed the CYBER 170 APL2 at a level earlier than 631, and desires to migrate APL applications to the CYBER 180, then PSR AP2A091 must be installed in APL2 prior to conversion of APL2 workspaces.

9.2 Manual Errata

The following items are Manual Errata in the area of APL:

1. CCP to CDCNET Differences with APL

9.2.1 CCP to CDCNET Differences with APL

APL users who are accustomed to accessing APL under CCP, and are migrating to use of CDCNET, should be aware of the following differences from the description in the APL for NOS/VE Language Definition Manual (60485813) chapter 12:

a. Network Command Character

Under CCP the default network control character is ESC, user break 1 is CONTROL/P, and user break 2 is CONTROL/T. Under CDCNET the default network command character is %, user break 1 is %1, and user break 2 is %2. If you change the code set to TPAPL or BPAPL, without having first changed the network command character, then the APL division symbol key will be seen by CDCNET as %. The consequence is that if you type the division sign followed by 2 as the only two characters of a line, this will be treated as user break 2, instead of the expected APL function, reciprocal of 2. APL users may find it convenient to change the network command character from % to the escape key. The command to do this is:

```
%CHANGE_TERMINAL_ATTRIBUTES NCC=ESC
```

Having made this change, if you then wish to have user break 2 be CONTROL/T, enter the command:

```
<esc>CHANGE_TERMINAL_ATTRIBUTES AC=DC4
```

where <esc> is the escape key.

If you wanted the network command character to be ESC, but forgot to change it before changing the code_set, you can type

```
<$dv>CHANGE_TERMINAL_ATTRIBUTES NCC=ESC
```

where <\$dv> is the division symbol key.

- b. Another important difference is in the method of telling the operating system that you are using the APL character set. The description on pages 12-4 and 12-5 of the APL manual (which was written for CCP use) works also when you are using CDCNET, except that instead of AR, use ACTAR for auto-recognition. If you get the diagnostic:

Auto-recognition not defined for this line.

then you must either find a line for which auto-recognition is defined, or use the CHANGE_TERMINAL_ATTRIBUTES command to change the code set. If you have changed the network command character to the escape key, <esc>, then the command to change the code set is:

<esc>CHANGE_TERMINAL_ATTRIBUTES CS=TPAPL

for a typewriter-paired APL terminal, or

<esc>CHANGE_TERMINAL_ATTRIBUTES CS=BPAPL

for a bit-paired APL terminal.

The commands suggested above can be combined, for example:

%CHANGE_TERMINAL_ATTRIBUTES NCC=ESC AC=DC4 CS=TPAPL

See the CDCNET Access Guide Manual (60463830) for a full description of the CHANGE_TERMINAL_ATTRIBUTES command.

Chapter 10 BASIC

10.1 Significant Problems

The following items are significant problems in the area of BASIC:

1. Symbolic Debugging Using SET_STEP_MODE

10.1.1 Symbolic Debugging Using SET_STEP_MODE

BASIC does not allow any symbolic debugging when a procedure has been entered with the command:

```
SET_STEP_MODE ON U=PROCEDURE
```

For example, if a DISPLAY_PROGRAM_VALUE \$ALL is performed after entering a procedure via the SETSM command, the following error will occur.

"The current stack frame is not yet initialized."

A workaround to this problem is that a breakpoint can be set on the first line of the procedure. This will allow symbolic debugging.

PSR BC8A074, DB8A339

10.2 Notes and Cautions

The following items are notes and cautions in the area of BASIC:

1. Do Not Use Concurrent OPENs

10.2.1 Do Not Use Concurrent OPENs

The use of concurrent OPENs of a file is not recommended because it may cause unpredictable results.

Chapter 11 Basic Access Methods (BAM)

11.1 Significant Problems

The following items are significant problems in the area of BAM:

1. PRESET_VALUE File Attribute
2. MINIMUM_RECORD_LENGTH File Attribute
3. Reporting of Boundary Conditions
4. FSPSOPEN_FILE Opens File Regardless of Error Condition
5. SET_FILE_ATTRIBUTE Sets Some Attributes Incorrectly
6. Files Implicitly Attached Are Not Implicitly Detached
7. File Label Attributes Cannot Be Altered After Creation

11.1.1 PRESET_VALUE File Attribute

Although the PRESET_VALUE file attribute can be set by the SET_FILE_ATTRIBUTE command, this feature does not work. New pages assigned to a file are always initialized to zeros.

PSR NVOB204

11.1.2 MINIMUM_RECORD_LENGTH File Attribute

File attribute MINIMUM_RECORD_LENGTH is not set properly for RECORD_TYPE=F files. MINIMUM_RECORD_LENGTH should default to MAXIMUM_RECORD_LENGTH if it is allowed to default, but it is currently defaulting to 0. This affects Sort/Merge when the sort key is omitted and the default is taken. The user should explicitly set MINIMUM_RECORD_LENGTH to the same value as MAXIMUM_RECORD_LENGTH.

PSR NVOD398

11.1.3 Reporting of Boundary Conditions

Skipping forward/backward by records and partitions does not always report the encounter of boundary conditions with the correct exception condition. The user should not depend on receiving the correct error condition when encountering a boundary condition.

PSR NVOD424

11.1.4 FSP\$OPEN_FILE Opens File Regardless of Error Condition

If a FAP (file access procedure) returns abnormal status to FSP\$OPEN_FILE during an OPEN of a previously non-existent file, the system still creates the local file. For example, a user who makes a mistake in the initial OPEN of an indexed sequential file will receive a proper diagnostic, but the system will wrongly create the file anyway.

PSR NVOE150

11.1.5 SET_FILE_ATTRIBUTE Sets Some Attributes Incorrectly

The SET_FILE_ATTRIBUTE command for an existing file accepts the specification of a preserved attribute even if it conflicts with the present one. No error will be seen and the preserved attribute will not be changed. Use DISPLAY_FILE_ATTRIBUTES to verify which attributes were set. Also, a subset of preserved attributes may be changed with the CHANGE_FILE_ATTRIBUTE command.

PSR NVOC839, NVOD617, NVOD549, NVOD536

11.1.6 Files Implicitly Attached Are Not Implicitly Detached

Files implicitly attached are not implicitly detached.

PSR NVOE322, NVOE296

11.1.7 File Label Attributes Cannot Be Altered After Creation

Attributes in the NOS/VE file label cannot be altered after creation. This could affect some migrating applications using indexed sequential files because FILE_LIMIT could be updated on NOS or NOS/BE but not on NOS/VE.

RSE NVEA034

11.2 Notes and Cautions

The following items are notes and cautions in the area of BAM:

1. Circular Connections with COPY_FILE
2. Share Mode Messages

3. SET_WORKING_CATALOG Limitations

11.2.1 Circular Connections with COPY_FILE

COPY_FILE does not detect a copy of a file to itself via circular file connections. Since a circular copy will destroy the input file by writing over it, the user should be extremely careful to not create a circular file connection.

RSE NVEA118

11.2.2 Share Mode Messages

If you get an exception message of this form:

```
--ERROR-- File X: You are required to allow sharing of  
READ,SHORTEN,APPEND,MODIFY,EXECUTE, not the intended sharing of  
READ,EXECUTE.
```

the file x is already attached to the job in a manner which allows other jobs to share for more access than was authorized by the command which failed. For example:

```
SET_WORKING_CATALOG $user  
ATTACH_FILE x access_modes=(read,execute) share_modes=all  
INCLUDE_FILE x
```

The INCLUDE_FILE command is not willing to share with a writer but the ATTACH_FILE command allows sharing for write. As a result, this message would be displayed:

```
--ERROR-- File X: You are required to allow sharing of  
READ,SHORTEN,APPEND,MODIFY,EXECUTE, not the intended sharing of  
READ,EXECUTE.
```

To solve this, detach file x, and either reattach it with different share modes, or allow an implicit attach by INCLUDE_FILE.

11.2.3 SET_WORKING_CATALOG Limitations

The SET_WORKING_CATALOG command has no effect upon files whose names are created within applications. For example:


```
SET_WORKING_CATALOG $user  
EXECUTE_TASK
```

and the task generates file name A, then A will be \$LOCAL.A.

Files created by CYBIL programs which use a 31 character local file name given to an FSP\$OPEN_FILE are not subject to the working catalog specification. Programs that wish the SET_WORKING_CATALOG to be honored should specify a complete path name by using CLP\$CONVERT_STRING_TO_FILE. Files referenced as parameters of commands (i.e., files specified at command level) will use the working catalog.

RSE NVEA027

Chapter 12

CDCNET

The information contained in this chapter pertains to CDCNET as used in NOS, NOS/VE and NOS-NOS/VE dual state systems. Those sections specific to NOS or NOS/VE only are so indicated. The remaining material applies to CDCNET in all three types of network installations.

12.1 Installation - NOS/VE Only

The following items describe installation of CDCNET in a NOS/VE system:

1. Initial Installation
2. Upgrade Installation

12.1.1 Initial Installation

The CDCNET portion of the initial NOS/VE installation process (described in chapter 1) defines a single terminal that you use to configure the remainder of the CDCNET network. The configuration process involves modifying the files in the `$$SYSTEM.CDCNET.SITE_CONTROLLED` catalog. Before you can modify these files, you must grant permission to users to update either the entire catalog or individual files within the catalog. The `REPLACE_CONFIGURATION_FILE` command requires that you grant both `CYCLE` and `CONTROL` permissions because of the method the command uses to update the files. You must grant these permissions from the system console.

```
create_catalog_permit c=$system.cdcnet.site_controlled g=user ..  
fn=family-name u=user-name am=(all cycle control)
```

??where "family-name" is xxx and "user-name" is xxx.??

Information on defining the network configuration appears in the CDCNET Configuration and Site Administration manual. Because of the limited number of site users who must use CDCNET host commands, they are not included on the default command list. Users who must use any of the CDCNET host utilities or maintain CDCNET configuration procedures should add the following command to their login prolog:

```
set_command_list a=$system.cdcnet.version_independent.command_library
```

12.1.2 Upgrade Installation

The upgrade to a new release or the installation of a Batch Critical Update (BCU) changes the CDCNET version level. The version level is a 4-digit hexadecimal number that is referenced as follows:

1. In the catalog structure of the CDCNET permanent files (that is, the `$$SYSTEM.CDCNET.VERSION_vvvv`, where `vvvv` indicates the version level). This catalog stores the files for the specified version level of CDCNET.
2. In the value for the `VERSION` parameter of the `DEFINE_BOOT_DEFAULTS` command. This command is specified in the CDCNET exception list file (`$$SYSTEM.CDCNET.SITE_CONTROLLED.EXCEPTION_LIST`). The value of the `VERSION` parameter controls the default level of the software to be loaded into all of the CDCNET Device Interfaces (DI's).
3. In the value for the `CDCNET_VERSION` variable stored in the `$$SYSTEM.CDCNET.VERSION_INDEPENDENT.CDCNET_VERSION` file. This value controls the default level of the three CDCNET host utilities, which are `ANALYZE_CDCNET_DUMP` (ANACD), `MANAGE_CDCNET_CONFIGURATION` (MANCC), and `NETWORK_PERFORMANCE_ANALYZER` (NPA).

Please note that not all CDCNET permanent files are stored under the `$$SYSTEM.CDCNET.VERSION_vvvv` catalog. CDCNET configuration files are contained in the `$$SYSTEM.CDCNET.SITE_CONTROLLED` catalog. These are the files that you must modify to describe the CDCNET configuration at your site. CDCNET software files that are version independent (that is, used by all versions of CDCNET software) are contained in the `$$SYSTEM.CDCNET.VERSION_INDEPENDENT` catalog.

When you upgrade to a new release or install a correction package, `UPGRADE_SOFTWARE` (UPGS) automatically updates the default version level of the CDCNET host utilities to the new level. UPGS also installs new copies of the files in the `$$SYSTEM.CDCNET.VERSION_INDEPENDENT` catalog. To complete the installation, you must do the following:

1. Update the newly installed files to reflect any modifications that you made to the previous versions of files in the `VERSION_INDEPENDENT` catalog. The new files are in the `$HIGH` cycle, and the previous versions of the files are in the `$HIGH` minus 1 cycle.
2. Update the CDCNET exception list to the new level and reload your DI's. You can use the `SET_VERSION_LEVEL` command (described later in this chapter) to change the exception list.
3. Delete any versions of CDCNET software that your site is no longer using. To do this, execute the commands shown below from the NOS/VE system console. Before executing the commands, be sure that your CDCNET exception list (file `$$SYSTEM.CDCNET.SITE_CONTROLLED.EXCEPTION_LIST`) and CDCNET version file (file `$$SYSTEM.CDCNET.VERSION_INDEPENDENT.CDCNET_VERSION`) do not reference the

version of CDCNET you intend to delete. In a multi-host network, you should also be sure that no other host will request a load of the deleted level.

```
task ring=7
delete_catalog_contents $system.cdcnet.version_vvvv
delete_catalog $system.cdcnet.version_vvvv
taskend
```

where "vvvv" represents the version level of CDCNET that you wish to delete.

12.2 Significant Problems

The following items describe significant CDCNET problems:

1. L664 Problems Corrected
2. Dump Analyzer (ANACD) Deficiencies
3. MANCC Deficiencies
4. Network Performance Analyzer (NPA) Deficiencies
5. Log Files in Date/Time Order
6. Automated Command Entry At Initial Connection May Fail.
7. Line Hangs
8. KILL SYSTEM DUMP Parameter not Honored
9. MDI/MTI Resets During Host Mainframe Maintenance - NOS/VE Only
10. Mismatching X.25 Network Identifiers can Reset DI
11. No Output From 53X Printer - NOS/VE Only
12. Congested MDIs - NOS and NOS/VE Dual State
13. Unrecognized NETOU Secondary User Statements - NOS Only
14. NOS Login - NOS and NOS/VE Dual State
15. No Disconnect for IAF Inactivity Timeout - NOS Only
16. NETFM Local File Name Change - NOS Only
17. Misleading A-A Connection Failure Reason Codes - NOS Only
18. First Time Installation Without FSE - NOS Only

12.2.1 L664 Problems Corrected

The following CDCNET problems documented in the L664 SRB have been corrected:

1. NOS 2.4.3 user login into NOS/VE 1.2.1

A NOS 2.4.3 user connected to CDCNET 1.0 will hang when attempting to log into a NOS/VE 1.2.1 System (VEIAF). Note that this is a back level PSR support combination. This problem (PSR AC1E628) has been corrected in a later level CDCNET Release 1.0 BCU (090B). Check Solver for an installation bulletin or contact Central Software Support for more information.

2. Network Performance Analyzer (NPA) Deficiencies

- . Log file data that is statistical in nature is condensed by NPA REFCLF into statistical intervals (typically one hour) and recorded to the appropriate statistical data bases. On NOS/VE, the data for the last two statistical intervals for these data bases will not be reflected on the corresponding reports.

PSR AC1E567

- . On NOS, REFCLF may abort with the following message when processing very large log files (>12000 PRUs):

NO MEM FOR NEW STCK SEG

If this occurs, reissue the identical command successively until a normal termination is achieved. REFCLF will bypass the log records already processed and continue from the point of the abort. It is important to continue this process until normal termination. If not, a NPA data base may be left in a condition that will cause a subsequent CRECAR command to abort with a CRM ERROR.

PSR AC1E631

To avoid this problem, terminate the current log file more frequently to reduce the resultant log file size. Another alternative is to utilize the REFCLF time and date parameters to bracket a smaller segment of a large log file.

PSR AC1E583

3. Comments on Configuration/TDP/TUP Files

The command parser in CDCNET incorrectly diagnoses a statement error under some circumstances where comments are the only constructs on a line and where leading spaces precede the initial quotation symbol. To avoid this problem, put the initial quotation symbol in column 1 of all comment-only lines.

PSR AC1C972

4. Hold Page Does not Work - NOS/VE Only

In certain cases for a terminal connected to NOS/VE, hold page will not be processed when output consists of lines of exactly PAGE_WIDTH (PW) characters to be output.

5. Invalid Routing Tables Possible - NOS/VE Only

A Routing Information Data Unit (RIDU) which shows an added or deleted network (lowest numerical network ID), or the first RIDU from a particular DI may be ignored in a receiving DI based on the value of an uninitialized variable. The consequence of this is invalid Routing

tables which makes it impossible for the DI to communicate with some other DIs or a NOS/VE host. Since only MDIs to NOS/VE hosts generate RIDUs, this can only happen to TDIs in NOS/VE environments. The effect is that the TDI cannot communicate with the host, and command processors (kept on the host) cannot be loaded. A TDI on loading might reset with reset code 'NO_CONFIGURATION_FILE_OBTAINED' 1B(16). A DI dump file can be examined to determine if any of the above situations have occurred.

There is no guaranteed workaround. However, MDIs between NOS/VE hosts and Ethernet LANs are the only systems which generate RIDUs. Making the network ID of the channelnet between the MDI and the host numerically higher than the Ethernet (as in default CDCNET configuration files) will make this problem less likely to occur. If the above mentioned reset occurs, the TDI may initialize successfully the next time. Once the catenet is stable, this problem cannot occur.

PSR AC1E632

6. Incorrect Processing of FN 57 and FN 58 - NOS Only

The CTRL/CHAR/R message is rejected by CDCNET when FNs 57 and 58 are requested with zero values. This is the recommended method for deselecting transparent message forwarding on message length. However, until this problem is fixed, these FNs should not be requested with zero values. In order to deselect transparent forwarding on message length, send the CTRL/CHAR/R message without specifying FN 57 or FN 58.

7. Batch Device Accounting Message - NOS Only

The SCTE C2, C4 and C5 NOS Account File messages are not correct for batch devices. Incorrect information is reported.

PSR AC1E444

12.2.2 Dump Analyzer (ANACD) Deficiencies

1. There are two problems associated with the DISPLAY_DATA_QUEUE subcommand.

- a. ANACD does not terminate if the circular queue LSA_PRESERVE_QUEUE is displayed with DISPLAY_DATA_QUEUE. If this queue must be inspected either use DISPLAY_MEMORY or use DISPLAY_DATA_QUEUE interactively only and terminate with a user terminate break when sufficient output has been generated.
- b. On NOS/VE only, if an output file is specified on the DISDQ subcommand then the buffer chain part of the display will not go to the designated file but to a file with a name of the form \$nnnnnn in the current working catalog. The nnnnnn used to form the name of this output file is taken from the first six characters of the system allocated local file name.

2. When running under the NOS/VE operating system with the output of the \$ERRORS file directed to another file via the CREATE_FILE_CONNECTION (CREFC) command, then only the last ANACD error message is saved on the connected file. To receive all error messages for a Dump Analyzer session in an external file, the output of both the \$ERRORS and \$RESPONSE files should be directed to the same external file, specifying a file position of \$EOI as part of the file reference. The following example connects standard files \$ERRORS and \$RESPONSE to file \$LOCAL.DAERRORS. The file is positioned to the end of information prior to its use.

```
CREFC $ERRORS      $LOCAL.DAERRORS.$EOI
CREFC $RESPONSE    $LOCAL.DAERRORS.$EOI
```

12.2.3 MANCC Deficiencies

1. When CRECF is used to create a system configuration file for a TDI, a DEFINE_ETHER_NET (DEFEN) and a DEFINE_ETHER_TRUNK (DEFET) command are included in the configuration file; in both commands, the TRUNK_NAME (TN) is given a default name of ETHERNET1. Neither of these commands is required in a TDI system configuration file, as they are performed implicitly by the TDI software load process. The presence of these commands in the configuration files does not adversely affect the loading and configuration process. However, log messages and operator alarms may be issued during configuration indicating that the trunk and network are "already defined". If these commands are present in a TDI configuration file, it is recommended that network administration and operations personnel be advised that these log messages and alarms are not a cause for concern. Additionally, the implicit trunk and network definition by the load process assigns a trunk name of \$ESCIx, where x is the slot number in the DI that contains the ESCI card, and the network name of \$NET_xxxxxxx, where xxxxxxxx is the ETHERNET network identifier. These names should be used, where needed, on any commands in the TDI configuration file or on any commands that mention this ETHERNET trunk or network solution name. Note that both the DEFINE_ETHER_NET and DEFINE_ETHER_TRUNK commands are required in an MDI configuration file, as the MDI load process does not implicitly perform the network or trunk definition.
2. The same problem described above also applies to both TDI and NDI files created in the EDICF environment using the TDI and NDI configuration templates.
3. When running under the NOS/VE operating system and the output of the \$ERRORS file is directed to another file via the CREATE_FILE_CONNECTION command, then only the last MANCC error message is saved in the connected file. To receive all error messages for a MANCC session in an external file, the output of both the \$ERRORS and \$RESPONSE files should be directed to the same external file, specifying a file position of \$EOI as part of the file reference. The following example connects standard files \$ERRORS and \$RESPONSE to file \$LOCAL.NCERRORS. The file is positioned to the end of information prior to its use.

CREFC \$ERRORS \$LOCAL.NCERRORS.\$EOI
CREFC \$RESPONSE \$LOCAL.NCERRORS.\$EOI

12.2.4 Network Performance Analyzer (NPA) Deficiencies

1. NPA on NOS/VE does not produce the accounting data base NPBACNT. No data is written if ACNT is selected when reformatting log files. This will be provided in a future release. Also see Operational Considerations Section for additional NPA information.

PSR AC1D353

2. The NOS/VE NPA subcommand RELOAD_NPA_DATA_BASE does not work.

PSR AC1D354

3. Under certain circumstances on NOS/VE, NPA reports SFTWRP3, EVNTRP3 and ETHRRP2 cause a CRECAR command error 'Invalid BDP Data'. The reports are not produced when this condition occurs.

PSR AC1E569

When executing REFCLF on NOS/VE, you must specify the complete file path name of the log file to be processed, it will not process the \$SYSTEM.CDCNET.LOG file by default. If the log file specified contains multiple cycles, you must supply the cycle numbers to REFCLF (in ascending order, from \$LOW to \$HIGH) in order to have the log files processed in the proper order. By default, REFCLF only processes the highest cycle.

PSR AC1E622.

4. REFCLF on NOS/VE does not correctly process multiple log files on a single run. Until this problem is corrected, specify only one log file per REFCLF execution.

PSR AC1E516

12.2.5 Log Files in Date/Time Order

REFCLF uses an incorrect assumption that all log files are in increasing date and time order. Because of this assumption, REFCLF can reprocess data that it has previously processed resulting in misleading reports. This may also cause some data to never be processed, resulting in misleading reports.

PSR AC1E105

12.2.6 Automated Command Entry At Initial Connection May Fail.

At initial connection to CDCNET once the line "You may enter CDCNET commands." has been received, the user is supposed to be able to enter commands and have them accepted by CDCNET. If a TUP is automatically invoked that sends several lines of output, commands entered too quickly may be discarded. The failure will be diagnosed with the message "Input discarded.". The message "Output discarded." may also be received. In the cases to date where this problem has been encountered, it has been worked around by delaying a few seconds after receipt of the "You may enter..." message before sending the first command. The particular failure that has been investigated involved 8 short lines of output from the TUP. In this case a 3 second delay in the PC based automatic login process was sufficient to work around the problem at 1200 baud.

PSR AC1E851

12.2.7 Line Hangs

If characters other than carriage returns are entered as part of the auto recognition sequence, the speed of the line may be incorrectly determined. If the line speed is incorrect, the line will often remain hung in the auto recognition active state, until the terminal is reset or the line is disconnected. The user will often realize what has happened if he/she believes he/she is trying to connect to CDCNET initially. However, if the user thinks he/she is still connected in the network, the user may get hung in this situation and not realize it. For example, assume a user is connected to a NOS/VE host at 19.2KB. The user leaves the terminal. The TDI to which the user is connected resets. The user returns to the terminal assuming it is still connected and enters "SCU" followed by a carriage return. The line will now be hung in auto recognition. A break key or a terminal reset/line disconnect will allow the user to auto recognize successfully.

If a user detects no response from his or her terminal when initially connecting to CDCNET and a terminal reset will not correct the situation, the Network Operator should perform the following steps:

1. Verify that the line is configured correctly.
2. Enter DISLS to check the state of the line. If the state of the line is 'auto-rec active' or 'enable', then try the following sequence to clear the condition:
 - . Stop the line (STOL) and start the line (STAL);
3. If the STOL/STAL does not clear the line, then run on-line diagnostics for that line (STAPT) or if other lines on the Line Interface Module (LIM) are free, run the on-line diagnostics for the LIM (STALT). If the on-line diagnostics fail, proceed as directed by the diagnostics.

4. If on-line diagnostics indicate no problems but the terminal still does not respond and the line state (DISLS) shows that the line is not "active", then run on-line diagnostics on the next lower number LIM on which no lines are active. Repeat this step until the line is cleared or the next lower number LIM has lines active.
5. If this process does not clear the line, then the line must be cleared by a MTI or TDI reset.

NOTE

This problem has occurred after a MDI or TDI has been reloaded with a new level of CDCNET.

12.2.8 KILL_SYSTEM DUMP Parameter not Honored

In a NOS/VE network, the KILL_SYSTEM DUMP parameter is not honored. Dumping of the MDI is governed solely by the EXCEPTION_LIST file. In order to use the KILL_SYSTEM DUMP parameter, the EXCEPTION_LIST file should be updated so that reset code 32(16) causes a DI dump and reset code 33(16) does not. Reset code 32 is the code generated as the result of KILL_SYSTEM DUMP=YES and reset code 33 is the code generated as the result of KILL_SYSTEM DUMP=NO.

Reset codes are selected/deselected for dump by using the ADD_DUMP_ON_ERROR and DELETE_DUMP_ON_ERROR parameters of the EXCEPTION_LIST file commands DEFINE_BOOT_DEFAULTS and DEFINE_EXCEPTION_SYSTEM.

PSR NV0J272

12.2.9 MDI/MTI Resets During Host Mainframe Maintenance - NOS/VE Only

Cyber 180 channel connected CDCNET systems (MDI or MTI) interfacing with NOS/VE Standalone mainframes may reset when the connected host is down for maintenance. Actions that have caused this to happen include mainframe power-down/power-up sequences or loading a new level of CIP firmware. Since the mainframe under maintenance is not actively connected to the network, there is no user impact or problems with other CDCNET devices or mainframes in the network when this problem occurs. The reset will result in a dump and reload of the channel connected Device Interface at the completion of NOS/VE deadstart and NAM/VE initialization on the mainframe that was down. The time the reset actually occurred will be recorded in the DI's System Configuration Table and can be accessed with the CDCNET Dump Analyzer command Display_System_Config_Table (DISSCT). This time can be used to confirm that the DI reset was the result of a maintenance action.

12.2.10 Mismatching X.25 Network Identifiers can Reset DI

A DI can reset when the Network Solution (NS) is configured with mismatching network_identifiers (the NI parameter on the DEFXN command).

12.2.11 No Output From 53X Printer - NOS/VE Only

A 53X printer used with the NOS/VE batch output facility may get into conditions where the printer stops or doesn't start printing. Three symptoms have been observed to cause these condition. They are described below along with two corrective actions to clear the condition:

1. A printer stops in the middle of a listing and nothing seems to be wrong with the printer. The batch device status display (from OPERate_Station) shows the device as active, the file transfer status as busy and a file assigned to the device.
2. A printer is not printing anything, but the batch device status display (from OPES) indicates that the file completion percentage keeps increasing.
3. A printer repeatedly gets a SENDER ERROR on one particular file at about the same place in the listing. After awhile the file may print on a different printer without any problem. The error can be seen by looking at the last lines of the listing. The last lines printed will indicate whether the transfer was completed normally or not. If not, the reason will be included.

These conditions can be cleared by one of the following methods:

1. Turn the printer power off, wait 2-5 seconds, and then turn the printer back on.
2. Use the Network Operator Utility (NETOU) to send the stop_line (STOL) and start_line (STAL) commands to the specific DI and line connected to the printer.

Either of these methods will clear the condition by disabling and re-enabling the line. The file assigned to the device will be queued again and printed later.

12.2.12 Congested MDIs - NOS and NOS/VE Dual State

During periods of peak loads, a NOS MDI may enter a buffer congested state. When this happens, the MDI will not accept any output from the host. Until the condition clears, all connections to the host will be affected, and no new connections will be accepted. Terminal users may see a delay in receiving their output. Responses to commands entered via NETOU may also be delayed because the commands are being held back in the host. This condition is temporary and allows the MDI to improve its buffer state. In cases of

extreme memory fragmentation and buffer congestion, the MDI may not be able to recover and should be reset if the condition lasts for more than five minutes.

12.2.13 Unrecognized NETOU Secondary User Statements - NOS Only

NETOU does not recognize secondary user control statements when processing EXECUTE_COMMAND_FILE commands. The default catalog for command files reverts to the user name specified at login. In addition, when requesting command files from alternate catalogs, if the file is private, the user name specified at login must have been explicitly permitted for the file.

12.2.14 NOS Login - NOS and NOS/VE Dual State

If a NOS login being entered through CDCNET is terminated abnormally prior to completing the entry, the ELP (END_LINE_POSITION) and EPP (END_PARTIAL_POSITION) terminal attributes are left set to NONE resulting in no line feed or carriage return for all future input. This condition can be corrected by entering a CHATA command (CHATA,ELP=LFS,EPP=CRS) or by completing a login.

PSR AC1D937

12.2.15 No Disconnect for IAF Inactivity Timeout - NOS Only

Non-working connections with output_action set to HOLD are not disconnected by the network when timed out by IAF. A side effect is that the CCT (CONNECTION_CONNECT_TIMEOUT) timer will not expire for switched lines causing the switched lines to remain connected after the connections associated with the line have all timed out. Users using multiple connections on switched lines should either hang-up at the end of their sessions or ensure that all connections have been disconnected before leaving their terminals. Sites may elect to limit the number of connections on switched lines to one to prevent this occurrence.

12.2.16 NETFM Local File Name Change - NOS Only

NETFM CREATE, UPDATE, and VERIFY commands cause NETFM to attach the permanent file being processed using a local name ZZZZZFM (instead of the actual filename as it did previously) in read mode. If a file is already attached in write mode (for example, a file was just defined), NETFM is not able to attach the file - resulting in a NETFM error. The write attached file should be returned before NETFM is called.

12.2.17 Misleading A-A Connection Failure Reason Codes - NOS Only

The reason codes reported to an application when an A-A connection attempt fails are misleading. This can cause difficulty in debugging the configuration definitions for A-A paths. In particular, a number of distinct connection failure conditions are being reported under reason code 146 (decimal). Unique reason codes will be provided in a future release.

12.2.18 First Time Installation Without FSE - NOS Only

Installation and operation of CDCNET requires the NOS Full Screen Editor. The instructions in the NOS IHB for installing CDCNET for the first time during a system upgrade (Chapter 3 and CDCNET section of Chapter 7) indicate that CDCNET should be installed while running the user's earlier version of NOS. If the earlier version of NOS does not contain FSE, the installation will fail. To work around this problem, delay the installation of CDCNET until NOS 2.5.1 has been deadstarted. (It is assumed that FSE is included with NOS 2.5.1 because it is required by CDCNET.)

This problem only applies to users who do not run FSE on the system from which they are upgrading.

12.3 Notes and Cautions

The following items describe CDCNET notes and cautions:

1. Multiple Interactive Connections
2. CREATE_CONNECTION Commands in Process
3. Non-Default Buffer Allocation Parameters
4. Adding X.29 Log Messages to Configuration Procedures
5. DI Configuration Files
6. CIM Configuration Limitations
7. LIM Configuration Caution for Synchronous Lines
8. Rotary Support
9. Condensed Statistics Removed from REFCLF
10. Accessing CDCNET Permanent Files - NOS/VE Only
11. Send_Command (SENC) Parameter Order - NOS Only
12. Procedure TC7TUP - NOS Only
13. CDCNET DI as only Network Hardware - NOS Only
14. 63 Character Set Not Supported - NOS Only
15. LCF Control Limitations for Auto Login - NOS Only

12.3.1 Multiple Interactive Connections

CDCNET 1.1 allows multiple connections per terminal with a released default of four connections per terminal. In general, a terminal which operates in a multiple connection environment requires more network resources than a terminal which does not. In particular, a terminal which maintains its non-working connections with an output_action of HOLD, causes additional

buffers to be used for holding output. Even if the connection is deleted by the host, the output is held until the connection is selected as the working connection. Although there is a limit on the amount of output that may be held, maintaining multiple connections in the state of holding output has the potential to degrade or deny service on active working connections. A site which plans to heavily utilize the multiple connection environment should monitor the memory and buffer states of the DIs in order to determine if additional memory is required. Also see section CDCNET R1.1 DI Memory Requirements for configuration information.

12.3.2 CREATE_CONNECTION Commands in Process

The CREC command does not complete until the selected service has accepted or rejected the users connection attempt. Under some conditions a service may not be able to immediately respond to a connection attempt. In extreme cases the response may not arrive for an extended period of time. While the CREC command processor is waiting for a response from the service, the users working connection is the connection from which the CREC command was issued and the user is put in solicited input mode. While in solicited mode, user input will be accepted by the network but will not be processed until the CREC command completes. The user is essentially locked out until the CREC command completes.

The user can enter the Break or Attention character during this time to cause input to be processed. A message received from the network operator during this time will also cause input to be processed. The input following either of these events will be treated as a command and discarded if the CREC command had been issued from the \$CDCNET_Command connection. If the CREC had been entered from an \$I/O connection, the input will be forwarded on that I/O connection and any output from that connection will be delivered to the terminal. In either case, until the CREC command completes, the connection from which the CREC command was issued remains the working connection and no terminal user commands will be processed.

12.3.3 Non-Default Buffer Allocation Parameters

For CDCNET 1.1, CDC strongly recommends that the released default settings for buffer sizes, buffer percentages, various memory and buffer congestion thresholds and stack sizes (specified on the DEFINE_SYSTEM (DEFS) command in the configuration procedures) be used. Some non-default settings of these parameters are known to cause problems.

12.3.4 Adding X.29 Log Messages to Configuration Procedures

No changes have made to the released terminal_user_procedures or the terminal_definition_procedures. However, the released configurator procedures have been modified for X.29 PAD. To run X.29, add the following command to the existing configurator procedures:

```
change_source_log_group add_message_number=(1340..1343)
```

12.3.5 DI Configuration Files

The released DI configuration files created during installation do not cause message number 74 to be logged. Add the following command to all configuration files after the initial define_source_log_group command:

```
change_source_log_group add_message_number=74
```

The configuration files produced using MANCC are also in error. Add the following two commands for logging and alarms to configuration files produced by MANCC:

```
change_source_log_group..  
  add_message_number = (74,550,551)  
define_source_alarm_message..  
  message_number = (550,551)
```

12.3.6 CIM Configuration Limitations

CDCNET 1.1 allows only one terminal protocol to be supported by a Communications Interface Module (CIM), i.e., ASYNC or HASP. In order to reconfigure a CIM, for example, from supporting ASYNC to supporting HASP, the DI configuration file must be updated and the DI must be reset.

12.3.7 LIM Configuration Caution for Synchronous Lines

The clocking parameter on the DEFL command has a default value of INTERNAL. This means that the LIM will generate the required clocking signals for transmit and receive (with NULL modem cable TN109). If the terminal provides the transmit clock, set this parameter to TRANSMIT (with NULL modem cable TN109). Note that most terminals will generate the transmit clock as defined by the RS232 standard. When using a modem, this parameter must be set to EXTERNAL, (and use modem cable TN108), since the modem will generate both clocking signals. Inappropriate selection of INTERNAL clocking can cause data to be received with errors or not at all.

12.3.8 Rotary Support

CDCNET is currently not capable of busyng out modems. If a communications line connected to a rotary is taken out of service, the rotary will become blocked, resulting in ring with no answer. If a line within a rotary must be taken out of service, the modem should be manually busied out. Refer to the modem documentation to determine how to do this.

12.3.9 Condensed Statistics Removed from REFCLF

1. DIOS, ETHR, HDLC, MCIS, SESS and TERM statistics are no longer collected into records (condensed) by REFCLF. These statistics are merely written to their respective databases. CRECAR will be changed in a future release to perform condensation.

NOTE

Since the these statistics are commonly written to the log files at one hour intervals, CRECAR reports will show few changes (except as noted in item 3).

2. The REFCLF parameter, SI (STATISTICS_INTERVAL), is accepted but is non-functional.
3. Actual ending times (HHMM) are reported by CRECAR in its DIOS, ETHR, HDLC, MCIS, SESS and TERM reports. Times are not rounded up to the nearest hour. Also, there are no **** entries for missing time intervals.
4. High/low limits for the previously mentioned CRECAR reports are frozen to reflect those for one hour intervals. Previously, the value specified for the SI parameter affected these limits.

12.3.10 Accessing CDCNET Permanent Files - NOS/VE Only

All CDCNET "data" files are private files under the NOS/VE \$SYSTEM family. Any user having a need to access any of the these files or catalogs must be granted explicit permission using the CREATE_CATALOG_PERMIT (CRECP) or CREATE_FILE_PERMIT (CREFP) commands. The following lists the catalogs/files in the \$SYSTEM.CDCNET catalog with a brief description:

Item	Description
SITE_CONTROLLED	Catalog of configuration files
DUMP	Catalog containing DI dumps
LOG	File containing network LOG files
ANALYSIS	Catalog of NPA database files from processed LOG files.

The CRECP and CREFP commands are documented in the NOS/VE System Interface Manual.

12.3.11 Send_Command (SENC) Parameter Order - NOS Only

The order of the parameters for the NETOU SEND_COMMAND (SENC) has been reversed for CDCNET 1.1. This change makes sending commands to the DI easier. For example, use:

```
send_command,command='disdss',system=a_di_system_name, or  
senc 'disdss' a_di_system_name
```

The command sent is now the first parameter and the system name has been moved to be the second parameter.

12.3.12 Procedure TC7TUP - NOS Only

The TC7TUP terminal user procedure has been changed to turn echoplex on. This is consistent with how CCP treats terminal class 7 terminals. The newly released TUPs, X364, CDC722_30, and VT100 also turn echoplex on. The TUPs PC_CONNECT_10, PC_CONNECT_11, and MAC_CONNECT_10, which are based on TC7TUP, leave echoplex off since that is the preferred mode of operation for Control Data CONNECT.

12.3.13 CDCNET DI as only Network Hardware - NOS Only

CDCNET accounting information, described in Section Accounting Information of this SRB, is written to the NOS account file by CS; additional types of information are written by CDCNET to log files created by NETLS. Therefore, even if no NPUs are present in the communications subsystem, CS must run if these account file messages are to be generated. CS will not initialize unless a network configuration file (NCF) is present. CDCNET-only systems will need to generate a dummy NCF to enable logging of NOS account messages.

A dummy NCF can be generated by using the following source input to NDLP.

```
NCFFILE : NFILE.  
END
```

The local file NCFFILE generated by NDLP should be made permanent as a direct access file of the same name under user name NETADMN.

If a CDCNET-only site does not want the NOS account file messages to be generated, then the CS job should not be submitted when NAM is started. This is achieved by changing the JOB statement for JOBCS in all PARAM records (INIT-MRECOV) in the NAMSTRT file under user name NETOPS to include the DI parameter. This change prevents the CS job from being submitted when NAM is initiated.

JOB(JOBBS,CS,SY,NS,DI) CS JOB.

If the CS job is initially disabled it is still possible to start the logging of NOS account file messages by a manual call to NAMI after NAM has been initiated.

NS is not required in a CDCNET-only configuration. The NAMSTRT JOB statement can be modified to include the DI parameter as follows:

JOB(JOBNS,SY,NS,DI) NS JOB.

Note that this change should only be done if no NPUs are present. This must be done in all PARAM records (INIT-MRECOV).

12.3.14 63 Character Set Not Supported - NOS Only

CDCNET does not support hosts running in 63-character mode.

12.3.15 LCF Control Limitations for Auto Login - NOS Only

Since CDCNET lines and terminals are not described in the NCF (as NPU lines are) it is not possible to supply auto login information in the LCF for a given terminal. It is possible, however, to specify auto login information for all terminals that access the host through a particular MDI/MTI. Include auto login information in the LCF for all of the terminal names that may be generated by the algorithm described in the Accounting Information Section of this SRB, entitled "NAM Terminal Name Generation."

12.4 Operational Considerations

The following items describe operational considerations for CDCNET:

1. NOS-NOS/VE MDI Differences
2. CDCNET R1.1 DI Memory Requirements
3. Access Procedures for ANACD, MANCC and NPA - NOS/VE Only
4. ANACD - Dump Analyzer
5. Flow Control
6. DI Clock Maintenance
7. NPA
8. Excessive Logging
9. Recommendations for Logging Configuration
10. NAM/VE Attribute Management Differences
11. Access Path Determination - NOS Only

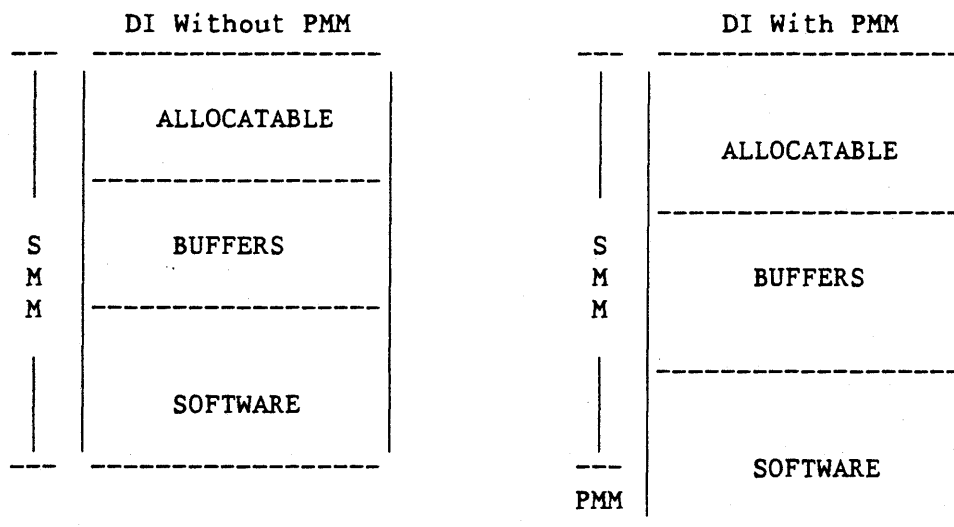
12.4.1 NOS-NOS/VE MDI Differences

NOS MDI/MTI configuration files can not be used for configuring NOS/VE MDI/MTIs. Specific details are described in CDCNET Configuration Site and Administration Guide (60461550).

12.4.2 CDCNET R1.1 DI Memory Requirements

1. Overview of DI Memory Usage

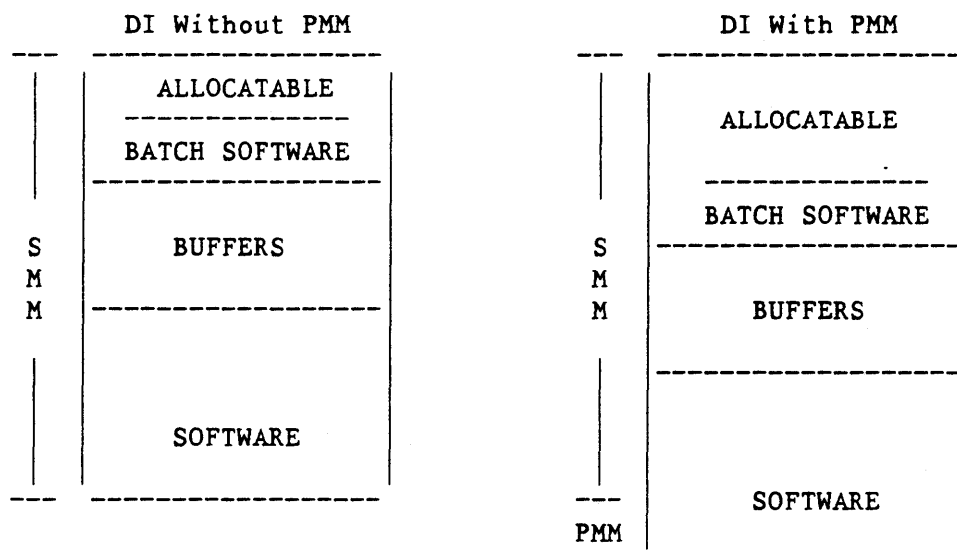
To understand the memory requirements for the supported DI configurations, it is useful to review how DI memory is utilized. After initialization of the DI, the software loaded is the basic network communication services needed by the DI, and the configuration file services specified in the configuration file. At the end of configuration, a contiguous block of memory is used by software, and the remainder of memory is free for other use. The DI system divides the free memory into two parts: buffers, and allocatable memory. Buffers are used to hold the messages for terminal users and for other services needed by the DI. Allocatable memory is used for tables, and for the loading and execution of additional software (e.g., command processors and software for services started on demand), and for tables used by network management functions. After the division of free memory, DI memory appears as shown below.



The basic memory requirements for DI software are about 500K bytes. For a TDI, the basic requirements, include the requirements for interactive services and the Asynchronous TIP. A 1 MB DI without PMM has about 250K bytes for buffers and about 250K bytes for allocatable memory. A 1 MB DI with PMM has about 315K bytes for buffers and 315K bytes for allocatable memory.

60K bytes of allocatable memory should be reserved for the loading of transient software and for network management functions. This leaves about 190K on a 1 MB DI without PMM for device configuration and connection tables and about 255K on a 1 MB DI with PMM. On a TDI, each configured device (e.g., asynchronous terminal or batch device) requires about 2K to configure. Each connection established for an interactive or batch user requires 4K bytes. A 1 MB TDI can support 32 terminals each with one active connection, i.e., $190K/(4K-2K) = 32$ connections.

A TDI with batch terminals requires an additional 35K bytes to load the batch processing software. The batch software is typically loaded as the communications lines for batch terminals and printers become active and after the basic configuration is complete. The memory to load the batch software is taken from allocatable memory, thereby reducing by 35K memory available for device configuration and connection tables.



The addition of batch services reduces the amount of allocatable memory to about 155K on TDI without PMM and 220K on a TDI with PMM (note that NOS PSU is handled as an interactive process rather than a batch process, and does not require the loading of batch software). On a TDI with the HASP TIP, allocatable memory required for software increases to 50K. Thus, the allocatable memory for device configuration and connection tables is reduced to 140K for a 1MB TDI, and to 205K on a 1MB TDI with PMM. Note that the same calculations apply to a NOS/VE MTI. A NOS MTI, however, is a different story because additional connection tables are needed to gateway CDCNET connections into NOS hosts. This increases the allocatable memory required for each active connection to 6K from the 4K seen on a TDI. A NOS MTI also requires 55K bytes more memory for basic software than a TDI. Thus, the allocatable memory for device configuration and connection tables is about 200K on a 1MB NOS MTI with PMM. A 1 MB NOS MTI with PMM will support 25 interactive terminals each with 1 active connection.

2. The following Guidelines summarize this analysis and demonstrate how they can be applied.

One Megabyte TDI Memory Support (1 SMM board, no PMM)

32 ASYNC lines can be configured with a maximum of 32 connections. For NOS, PSU Printers in place of ASYNC terminals do not increase memory usage.

Guideline 1 Each ASYNC Interactive terminal takes 2K bytes to configure, and an additional 4K bytes for each active connection. Note that a line can be configured to allow multiple connections.

Guideline 2 VE Batch printers add 35K bytes beyond the 1 MB TDI support. If VE Batch printers are to be configured on a TDI, fewer lines (2K bytes each) must be configured, or fewer connections must be allowed. An alternative is to add memory with either PMM or SMM. Added memory only counts 50% towards DI terminal buffer availability. PMM adds 64K bytes, SMM adds 512K bytes towards DI terminal buffers.

Guideline 3 HASP I/O Stations add 50K bytes beyond the 1 MB TDI memory requirement. If HASP I/O Stations are to be configured on NOS or NOS/VE, fewer lines or fewer connections must be configured. An alternative is to add memory with either PMM or SMM. If both HASP I/O Stations and ASYNC printers are configured for NOS/VE, add 50K bytes beyond the 1 MB Memory requirement (ASYNC printers do not add any additional memory requirements beyond HASP). Each active HASP I/O Station device takes an additional 6K bytes. A minimum HASP I/O Station has two devices, a printer and a console. Therefore, a minimum HASP I/O Station will require 12K bytes when it is active.

Example 1 TDI with 21 ASYNC terminal (1 connection each), and 1 HASP I/O Station with two devices could be supported in a 1 MB TDI. Memory usage using Guidelines 1 & 3 is:

ASYNC Terminal Savings = $(32 - 21) * 6K = 66K$ bytes
HASP I/O Station Addition = $50K + 12K = 62K$ bytes

Example 2 NOS/VE TDI with 25 ASYNC terminals (1 connection each), & 1 ASYNC printer is supported in a 1 MB TDI. Memory usage using Guidelines 1 & 2 is:

ASYNC Terminal Savings = $(32 - 25) * 6K = 42K$ bytes
ASYNC Printer Addition = $35K + 6K = 41K$ bytes

EXAMPLES for TDI MEMORY SUPPORT

CONFIGURATION	ASYNC TERMINAL	ASYNC with HASP	VE ASYNC with ASYNC Printers
1 Megabyte TDI	32 ASYNC 32 Connections	21 ASYNC, 1 HASP, 22 Connections	25 ASYNC, 1 ASYNC Printer, 26 Connections
1 Megabyte TDI with PMM	32 ASYNC 48 Connections	28 ASYNC, 2 HASP, 30 Connections	Can intermix ASYNC & Printers 39 Connections
2 Megabyte TDI (2 SMM boards)	32 ASYNC 96 Connections*	28 ASYNC, 4 HASP, 96 Connections*	Can intermix ASYNC & Printers 96 Connections*

ASYNC - ASYNC Terminal (9.6K bps)
ASYNC Printer - CDC 536 Printer (9.6K bps)
HASP - HASP Station (9.6K bps) with two devices
* - Current testing limit is 96 connections

3. NOS MTI Memory Configuration (minimum MTI with 1 SMM, one PMM)

Up to 25 ASYNC lines with 1 connection per line can be configured. If additional ASYNC lines are to be configured, use Guideline 1. If HASP I/O Stations are to be configured, use Guideline 3.

4. NOS/VE MTI Memory Configuration (minimum MTI with 1 SMM, one PMM)

Up to 32 ASYNC lines with 1 connection per line can be configured. If additional ASYNC lines are to be configured, use Guideline 1. If HASP I/O Stations are to be configured, use Guideline 3. If ASYNC printers are to be configured, use Guideline 2.

5. NOS MDI Memory Configuration (minimum MDI has 1 SMM with PMM)

Up to 45 active connections can be supported (2K bytes are used per connection). If additional connections are required, additional SMM will be needed. Note that 2 SMM boards could support up to 300 connections from a memory standpoint. However, DI processor limits may be reached at less than 300 connections.

6. NOS/VE MDI Memory Configuration (minimum MDI has SMM only)

There are no NOS/VE MDI active connection memory constraints for the committed configurations.

7. NOS or NOS/VE NDI Memory Configuration (Minimum NDI has SMM only)

There are no memory constraints with up to two X.25 trunks.

12.4.3 Access Procedures for ANACD, MANCC and NPA - NOS/VE Only

ANACD, MANCC, and NPA operate as command utilities on NOS/VE. Because the commands used to invoke the utilities are SCL procedures (which execute lower level CYBIL programs), a problem can occur if SCL variables are referenced by commands inside the utility. This can be of particular importance when a utility is invoked from a procedure where status variables will commonly be used to control procedure flow.

The following sequence will not work:

```
create_variable local_status k=status
npa
create_databases status=local_status
```

"An error is generated at this point because..
local_status is unknown inside the utility."

To gain access to SCL variables inside the CDCNET host utilities, use one of the following methods:

1. To reference a variable created before the utility is invoked, the variables must be declared outside of the call to the utility with a scope of XDCL or JOB. The variables should be declared again from inside the call to the utility with a scope of XREF. This allows access to the variables from within the utility. For example:

```
create_variable local_status k=status s=xdcl
npa
create_variable local_status k=status s=xref
create_databases status=local_status
'Process the local_status variable.'
quit
```

2. To use a variable solely from within the utility, it must be declared after the command utility has been invoked, rather than outside of the utility. For example:

```
npa
create_variable local_status k=status
create_databases status=local_status
'Process the local_status variable.'
quit
```

12.4.4 ANACD - Dump Analyzer

The version or software release level of the dump to be analyzed should be as close to that of the ANACD software as possible, otherwise the results of some subcommands may be unreliable or provide incomplete data. This occurs because many ANACD subcommands examine DI software structures which may change from one version to the next as new features are added. Subcommands such as `DISPLAY_MEMORY` or `DISPLAY_LINKED_LIST` which do not rely on DI software structures will work on any dump.

Periods are no longer allowed to terminate ANACD subcommands. Periods must be removed from the end of the subcommand lines in ANACD directives files or a parser error will result.

Now that Dump Analyzer can run on NOS/VE, long file names (up to 31 characters) are accepted as valid on subcommands issued within the utility. A warning message is issued on NOS if a long name is used on a subcommand that only the first 7 characters will be used to determine the identity of the file. This means that on NOS it is the user's responsibility to ensure that file names used within ANACD are unique in the first seven characters. The advantage is that directive files containing subcommands can utilize long name files and be used on NOS and NOS/VE without change provided that names with special characters are not used.

12.4.5 Flow Control

Most terminals and microcomputers which allow use of 19.2 Kbps or faster lines require the use of downline flow control since they cannot accept sustained line traffic at full data rate. Also, some terminals/computers have problems at line speeds as low as 4800/9600 bps. Some terminals/computers automatically enable flow control; for others it is a configuration or run-time option. CDCNET defaults `CHARACTER_FLOW_CONTROL` (XON/XOFF handshaking) to ON to avoid flow control related problems, since they often result in apparently hung terminals or lines. While it is unlikely that this will cause any problems for most users, some terminals or protocol-transfer utilities may need to be able to send/receive all 256 characters. Site and user prologs and procedures which specifically disable flow control should be reviewed to determine if this is necessary. CDC strongly recommends that flow control be enabled whenever possible to avoid garbled or missing data or seemingly hung terminals. Users may require instruction on terminal/computer setup. Some programs, such as full screen products, enable flow control and leave it set upon exit; others, such as PFTF when processing XMODEM protocol transfers, disable it. To check on the setting of flow control, use the `DISPLAY_TERMINAL_ATTRIBUTES` command. If flow control is set incorrectly, it can be changed by using the `CHANGE_TERMINAL_ATTRIBUTES` command.

CDCNET supports control signal flow control (sometimes referred to as "out of band" or EIA) for input data, as an option that can be specified via the `EIA_FLOW_CONTROL` (EFC) parameter on the `DEFINE_LINE` (DEFL) command in the configuration file. For output, Clear_To_Send (CTS) flow control is always

enabled. If CDCNET needs to regulate the flow of input to avoid data loss and control signal flow control has been specified for the device, CDCNET will drop the Request_To_Send (RTS) signal output from the LIM (which is also input to the connected device). This stops the flow of input data from directly connected devices that use this signal for flow control. If control signal flow control is used in a directly connected DTE-to-DTE null modem configuration (e.g. terminal cabled directly to LIM), the TN472 or YA306 cable must be used.

Some terminals claim to support high line speeds and flow control but in fact cannot keep up with sustained high data rate traffic. Problems resulting from data overrun can include many symptoms, such as "garbage on the screen", "terminal locked up", partial loss of function key data after the host sends an X-OFF downline, valid data written to incorrect places on the screen, etc. Operations that take a long time such as screen clear often aggravate these situations.

It should be noted that in some cases it is necessary to reset the terminal in order to correct these problems. This would typically be the case with a terminal such as a 721 where there is heavy use of downline escape sequences. The loss of portions of an escape sequence can result in a very confused terminal.

It should also be noted that problems in this area may be encountered even when line speeds are the same under CDCNET as with CCP or other networks, due to timing and output data rates.

If a user line or terminal appears to hang, it may be possible to resume normal operation by manually entering X-ON, BREAK, or BREAK followed by X-ON; if these fail to correct the problem, resetting the terminal may be required. Using a lower line speed may help determine if the problem is line speed related; if it is, use a lower line speed until the problem is resolved. Also, verify that both the terminal and CDCNET have flow control enabled and that the modem, multiplexer, port selector, local area network, etc. are not "discarding" this flow control.

There are a number of conditions which can cause a terminal not to send an X-ON after an X-OFF has been sent to the network. When this occurs, it appears as though the line is hung. Entering an X-ON from the terminal keyboard may resolve the problem. Two examples of when this situation occurs are as follows:

1. A reset code is sent to the terminal. On Zenith Z19/Z29 terminals, there is a sequence that resets the terminal. If this is sent, the terminal resets itself and forgets that it has sent an X-OFF. This condition has been encountered when some TDU files were doing this type of thing.
2. Entering of a Break on a 721 terminal sometimes results in the line being left in an X-OFF state.

The "emergency escape sequences" (Break/Control-A/Control-T and Break/Control-A/Control-X) do not work on some CDC 721 terminal configurations because some characters, such as Control-A, have special meaning to a CDC 721 terminal.

CDC 721 terminals with Revision 3.0 firmware have serious deficiencies with flow control. CDC recommends that these terminals with Rev 3.0 firmware not be used above 9600 bps. CDC 721 terminals with Revision 4.0 firmware also have serious problems with flow control at line speeds above 9600 bps. CDC does not advise use of a CDC 721 terminal in 132 column mode at line speeds above 9600 bps. Also, some users may see problems in 80 column mode at line speeds above 9600 bps. A ROM pack option is planned to address these problems. These problems have been seen on both NOS and NOS/VE systems, but appear to occur more frequently on NOS/VE screen mode operations than on NOS screen mode operations. In non-screen mode, the problems will most likely appear with equal frequency on either system.

12.4.6 DI Clock Maintenance

The value of the DI date and time in MPB battery backed RAM is initialized to 01/01/86 00:00:00 when first loading an R1.1 system, or on subsequent resets of an R1.1 system in which power has been lost to the MPB.

If a DI is brought down and not reinitialized for a period exceeding 24 hours, the automatically adjusted date will be wrong. For a DI which does not contain the network master clock, this will cause a few log messages to be issued with an incorrect date until the DI can automatically synchronize its clock with the network master clock.

If the master clock is being maintained in a DI, i.e. a NOS network, and if the master clock DI is initialized after being down for a period of 24 hours, the proper date and time should be adjusted in the master clock DI using the SET_DATE_AND_TIME command. All other DIs should then be synchronized using the SYNCHRONIZE_CLOCK command.

The date and time of last DI reset is derived from the DI date and time kept in MPB RAM. After a reset to load an R1.1 system or after a reset in which power has been lost to the MPB, the date and time of last reset as displayed by the display_system_status command, will be incorrectly reported as 86/01/01 00:00:xx. In order to properly initialize the date and time of last reset, the DI date and time should be correctly initialized and then the DI should be reset a second time.

12.4.7 NPA

Attempts to reformat CDCNET 1.0 log files with NPA on CDCNET 1.1 will produce unpredictable results.

Care must be taken when reformatting log files with REFCLF to avoid loss of data. When a log file is reformatted to a database, NPA records the time of the last log message into the database. On a subsequent reformat, any log message with a time earlier than the time recorded in the data base will be skipped. Reformatting two log files separately but in the wrong order is an example of a lost data situation. Reformatting the same two log files in a single NPA reformat run would be okay, as NPA will sort out the order.

12.4.8 Excessive Logging

Some log messages may occur quite often and cause problems in reviewing the NPA files or in using too much disk space. Sites are encouraged to disable additional log messages as necessary via the `DEFINE_SOURCE_LOG_GROUP` and `CHANGE_SOURCE_LOG_GROUP` commands. Two such messages to consider are:

1. 210 - an indication of a modem/line disconnect. Due to changes from AC10086, this message could be issued every 30 seconds for a failing modem or line. If this log message occurs frequently, the site should correct the hardware problem or disable the log message.
2. 283 - may occur as a result of an ESCI reset. If the "from status" and the "to status" (`network_up` to `network_up`) are both zero, CDCNET is still operational as indicated in the log message explanation, and the log message can be ignored. However, if the status is not zero, this message may indicate a different problem and should be investigated.

When the condition has been corrected, the log message can be re-enabled.

12.4.9 Recommendations for Logging Configuration

CDC recommends that sites define one primary and at least one backup log recorder whenever possible. If a site chooses to define multiple log groups, one primary and at least one backup log recorder should be defined for each log group.

A NOS/VE system can be configured as a primary log recorder via the `NOS/VE ACTIVATE_NETWORK_LOG (ACTNL)` command with the default priority (PRIMARY). A backup log recorder is defined with the priority BACKUP.

A NOS MDI or MTI can be configured as a log recorder via the `CDCNET DEFINE_RECORDER_LOG_GROUP (DEFRLG)` command. The primary log recorder for a NOS MDI or MTI is the one defined with the highest priority (the default and highest priority is 1). Backup log recorders are those defined with lower priorities.

The PRIMARY priority for NOS/VE log recorders is equivalent to the priority 1 for NOS MDI or MTI log recorders, and the BACKUP priority is equivalent to priority 255.

All NOS MTIs should be defined as log recorders with the default priority.

One NOS/VE system or one MDI connected to a NOS host should be defined as a primary log recorder for the network. If there are any other NOS/VE systems or NOS MDIs in the network, at least one of these should be defined to provide backup for the primary recorder. Whenever more than one backup log recorder is defined for a log group, a different priority should be specified for each. It is recommended that only one NOS/VE system be defined as a backup log recorder per log group since different backup priorities cannot be defined in this release. Redundant logging (multiple log recorders in the network defined with the same priority) is NOT recommended since it adds overhead to the DIs generating log messages (log sources), adds overhead to the network traffic, and increases permanent file space usage on the host.

When analyzing problems from DIs in a multi-host environment, log files from the hosts with backup log servers will need to be obtained. The log files may be reformatted and reports generated separately for each host or the log files from the backup host may be moved to the primary host so that a single set of NPA data bases and reports can be generated. If log files are moved to the primary host, log file names may need to be unique (log file names on the backup host may be the same as those on the primary host).

CAUTION: Care must be taken when reformatting log files from different hosts to avoid loss of data. Refer to the Operational Considerations Section on NPA for cautions on the use of REFCLF.

12.4.10 NAM/VE Attribute Management Differences

There are a number of differences between the CDCNET and NAM/VE attribute management commands, e.g. CHATA, CHACA, DISTA, DISCA, etc.

NAM/VE maintains a separate set of terminal and connection attributes for every instance of a terminal file open. As a result, the NAM/VE attribute management commands have a filename parameter to identify the target set of attributes to be changed.

In NAM/VE, connection attribute management is an application function and it is strongly suggested that connection attributes not be changed by terminal users.

CDCNET and NAM/VE use different syntax for specifying values for attributes that can have a value of a character, a list of characters, or a string of characters. The following table highlights the NAM/VE and CDCNET supported syntax:

SYNTAX	NAM/VE	CDCNET
'a'	Y	Y
'ab'	Y	N
ETX	N	Y
('a','b')	N	Y
'a'/'b'	Y	N
33(8)	N	Y
\$CHAR(33(8))	Y	N
↑c	N	Y

Because of the different syntax, the displayed values for attributes are different between CDCNET and NAM/VE. Also, NAM/VE integer values are displayed in decimal while CDCNET displays integer values in hexadecimal. NAM/VE utilizes lower case in its displays, while CDCNET uses upper case, exclusively.

CDCNET allows CRD and LFD values in the range 0..1000 and FFD in the range of 0..3000. NAM/VE allows these attributes in the range 0..999. When using NAM/VE if values for these parameters greater than 999 are required, the CDCNET command must be used to set them.

The NAM/VE attribute display commands have a `display_options` parameter for selectively displaying attribute values. CDCNET's DISTA and DISCA commands have parameters `terminal_attributes` and `connection_attributes`, respectively, for selectively displaying attribute values.

12.4.11 Access Path Determination - NOS Only

As described in the Accounting Information Section of this SRB entitled "NAM Terminal Name Generation", the terminal name provided to NAM by CDCNET is generated by an algorithm that does not include the physical port number. Therefore, with CDCNET terminals it is not possible to identify the exact physical path traversed by a given user to access the system, as it is for CCP. From an operational standpoint, this means that the NVF display of user names and the associated terminal name will no longer be useful for identifying the exact physical path over which a given user is accessing the system.

12.5 Accounting Information - NOS Only

The following items describe accounting information for CDCNET:

1. NAM Terminal Name Generation
2. Account File Messages

12.5.1 NAM Terminal Name Generation

CDCNET terminal names are not defined in the NCF. CDCNET MDIs/MTIs supply NAM with a terminal name of the following form:

Tccmmnnn

where: cc = logical link coupler node number
mm = logical link MDI/MTI node number
nnn = number in the range 1..255 for uniqueness

12.5.2 Account File Messages

The CDCNET terminal name generation algorithm affects sites that use the ABAP message from the account file to generate port utilization statistics. Any programs that process the account file to extract this type of statistical information should be modified to use the SCTE message. (The SCTE message is generated when a user logs out.) Any programs that generate user billing based upon correlation of the terminal name field in the ABAP message with the NCF to determine the characteristics of the line also need to be modified.

The format of the new account file messages is as follows:

Terminal Accounting Messages

SCTE, C1, NAM_terminal_name
SCTE, C2, device_type.
SCTE, C3, destination_service
SCTE, C4, di_name_or_address.
SCTE, C5, lim,port,vvvv. (vvvv=0)
SCTE, C6, tttttrrrrrssss.
SCTE, C7, ccccc.
SCTE, C8, connect_time.

Application-Application Accounting Messages

SCAP, C1, source_service. (always blank)
SCAP, C2, destination_service.
SCAP, C3, di_name_or_address. (system ID of remote DI)
SCAP, C4, mmm,prt,vvvv. (mmm=0, prt & vvvv=0 unless X.25)
SCAP, C5, tttttrrrrrssss.
SCAP, C6, ccccc.
SCAP, C7, connect_time.

where:

cccccc = Characters received
lim = Line Interface Module
mmm = LIM or mainframe channel interface
prt = Port number on the LIM
rrrrrr = Blocks/packets received
ssssss = Characters transferred
tttttt = Blocks/packets transferred
vvvv = X.25 virtual circuit number

12.6 Application/Terminal User Information

The following items describe application/terminal user information for CDCNET:

1. Host Unavailability Processing
2. Hi-Speed Micro-Mainframe File Transfers
3. Terminal Recognition Problems
4. User Breaks
5. Line Folding Changes

12.6.1 Host Unavailability Processing

Under certain conditions, it will take CDCNET software a relatively long time (minutes) to inform its users (connected to CDCNET via interactive terminals) about the loss of their connections to Cyber (NOS or NOS/VE) host applications, the inability to create new connections to Cyber applications, and the inability to execute terminal user procedures via the %D0 command.

12.6.2 Hi-Speed Micro-Mainframe File Transfers

Many personal/microcomputers cannot keep up with 38.4 Kbps traffic; some may even have problems at 19.2 Kbps and 9.6 Kbps. Protocol transfers via CONNECT, XMODEM, RMF, etc. should successfully detect and may be able to successfully recover from such errors, but non-protocol transfers at these speeds may be suspect.

12.6.3 Terminal Recognition Problems

Terminals with power supply problems, inaccurate bit clocks, or that are connected via very long cables may experience problems with auto recognition of line speed. These problems may occur with CDCNET where they have not occurred with other networks. The reason is that CDCNET auto baud detects over a much wider range of speeds than is supported by CCP and other

networks. The failure is because the character waveform is distorted. Even though problems are encountered with auto recognition, the terminal may run without problems if the line is configured explicitly for the desired line speed.

If the ASYNC TIP is not loaded as part of the initial DI load process, the first asynchronous user is likely to encounter a long delay (up to a minute in a heavily loaded system) before the terminal is recognized, due to the time it takes to load the ASYNC TIP dynamically. This delay can be avoided by including the command

LOAD_MODULE ASYNCTIP_MODULE

in each TDI/MTI configuration file. This does not have any negative impact on memory usage or allocation.

12.6.4 User Breaks

Since the User Break sequences for CDCNET differs from those of CCP, some user confusion can be expected. Once the newness wears off, the sequence of <ncc>1 for User Break 1 and <ncc>2 for User Break 2 should prove to be a benefit.

Note, however, that programs using transparent input may have "hard coded" recognition of the CCP User Break(s); FSE is an example of a program which has been "hard coded" to recognize control-T in screen mode.

12.6.5 Line Folding Changes

CDCNET 1.1 has changed its line folding algorithm so that when FL=OFF, lines of exactly PW width will cause one LFS to be omitted from the following line. This change was made so that output would not double space for terminals that fold after displaying character PW. However, terminals that delay folding until just before printing character PW+1 might now experience overprinting and may require FL=ON to work properly.

12.7 CDCNET Batch Devices

The following items describe CDCNET batch devices:

1. HASP Workstations
2. CDC 533-1, 536-1 Asynchronous Line Printers
3. Printing of the Banner Page Separator

12.7.1 HASP Workstations

1. CYBER 18 HASP Workstation

On the display terminal which provides the workstation operator with an interface to the system, the underscore character is mapped to a backspace character. To change the mapping so that the character is interpreted as an underscore, the following should be done:

CARD DECK SYSTEM:

Insert the following patch card before the HASP initiation (HWS) card:

PAT,1108/1803

↑

column 1

FLEXIBLE DISK SYSTEM:

Insert the following patch card before the HASP initiation (HWS) card using the procedure documented in the HASP RJE WORKSTATION CONTROLWARE INSTALLATION HANDBOOK (60475360) titled "Updating Configuration on Flexible Disk".

PAT,1108/1803

↑

column 1

2. CYBER 120 HASP Workstation

For correct banner page formatting, the default forms control specifications contained in :UTIL:FORMS:HASP_STANDARD may need to be modified. To change to an alternate forms control, refer to the Software Release Bulletin for the 553/556 HASP Emulator (SMD170474) under the 'Notes and Cautions' section. The following are the forms control settings for a 66 line page with the "automatic skip on perforation" printer switch disabled:

FORM LENGTH IN LINES PER PAGE

[66]

TOP OF FORM (CHANNEL 1) LINE NUMBER

[1]

BOTTOM OF FORM (CHANNEL 12) LINE NUMBER

[66]

VFU TAPE (1-66 CHANNEL NUMBER 2-11, OR NL)

[5-2, 6-3, 7-4, 8-5, 9-6,10-7,11-8,12-9,13-10,14-11,]

3. Printer Option Settings

The HASP workstation line printer main control panel options should be selected as shown in Table 6-5 in the CDCNET Batch Device Users Guide (60463863) with the following exceptions:

Option Number	Setting
21	Y/N
22	5
28	N
33	96/19
34	8

The TN109 cable should be used with the HASP workstation printer.

12.7.2 CDC 533-1, 536-1 Asynchronous Line Printers

See the CDCNET Batch Device User Guide (60463863) for information on suggested printer settings. The TN109 cable should be used with the asynchronous line printer.

12.7.3 Printing of the Banner Page Separator

In order for each banner page separator to be printed across the page perforations, the "automatic skip on perforation" switch on the printer must be disabled.

12.8 Manual Errata

The following items describe manual errata for CDCNET:

1. INPUT_OUTPUT_MODE Documentation
2. CDCNET Secure Terminal Procedure
3. HASP Workstation Configuration
4. Logging Configuration Command Changes
5. Log Message Changes
6. Command Continuation
7. Accessing CDCNET Utilities - NOS/VE Only
8. Retrieving a DI Dump File - NOS/VE Only
9. Terminal Class for Teletype Model 43 - NOS & NOS/VE Dual State
10. X.25 Configuration Recommendation - NOS Only
11. MDI Error Status - NOS Only
12. Conversion of Transparent Attributes - NOS & NOS/VE Dual State
13. Effects of Changing Terminal Classes - NOS & NOS/VE Dual State
14. CDCNET Mapping of NAM FN/FV Pairs - NOS & NOS/VE Dual State
15. Effects of NOS TRMDEF Command - NOS and NOS/VE Dual State Only

12.8.1 INPUT_OUTPUT_MODE Documentation

The Terminal Interface Manual (60463850A) incorrectly documents how the connection attribute INPUT_OUTPUT_MODE (IOM) operates. The following documentation should be used instead of the released documentation.

UNSOLICITED or U In this mode, CDCNET edits input as soon as it is received. The edited input is forwarded to the application when a forwarding character, such as ELC, is received. Output received while input is being edited is not sent until a forwarding character is received. Input received while output is active will suspend output until a forwarding character is received.

In this mode, each line of typed ahead input is edited and echoed to the terminal as it is entered.

SOLICITED or S In this mode, CDCNET edits and forwards input only when the application has solicited the next input. An application solicits input by sending a complete output message to the terminal.

When input has been solicited, the input editing and forwarding process, and the process of interrupting and resuming output, is the same as for the UNSOLICITED mode.

If the application has not solicited the next input, input is accepted by CDCNET but is not edited. No echoplexing, backspacing or cursor positioning will be seen at the terminal. User interrupts may be entered but will not be processed until the application solicits input. Output is transmitted as soon as it is received and received input does not interrupt output in progress.

In this mode, each line of typed ahead input is edited and echoed to the terminal only when it has been solicited by the application.

FULLDUPLEX or F In this mode CDCNET processes input and output independently of each other.

Input is edited as soon as it is received. The edited input is forwarded to the application when a forwarding character, such as ELC, is received.

Output is delivered to the terminal as soon as it is received.

12.8.2 CDCNET Secure Terminal Procedure

Any Terminal User Procedure (TUP) whose name begins with the 'commercial at' (@) character can only be executed when specified in the Define_Line command (DEFL). It cannot be executed via a DO command. This feature is most useful for TUPs that include logins (user validation) because the procedure cannot be executed by any terminal user.

12.8.3 HASP Workstation Configuration

CDCNET does not support the /*CONFIG directive to define the configuration of the devices on the workstation. Instead, the terminal operator enters the DO,<proc>,TDP command, where <proc> identifies the TDP containing the device configuration.

12.8.4 Logging Configuration Command Changes

CDCNET provides a site with the capability to define subsets of DI systems within the catenet and have all the systems within a subset transmit their messages to a common log file. Such a subset of systems is referred to as a log group.

In prior releases of CDCNET, the default and only log group name allowed was CATENET and therefore all systems within the catenet transmitted their messages to the same log file.

In this release, log group names other than CATENET are allowed. The log_group parameter on the DEFINE_SOURCE_LOG_GROUP (DEFSLG) command may be used to specify a site selected name for the log group that a CDCNET system is to belong to. A NOS MDI/MTI or a NOS/VE system within the catenet must be defined to provide the log recording function for the log group. In a NOS environment, the log_group parameter on the DEFINE_RECORDER_LOG_GROUP (DEFRLG) command is used to specify the name for the log group that a CDCNET system is to provide the log recording function for. In a NOS/VE environment, the groups parameter on the ACTIVATE_NETWORK_LOG (ACTNL) command is used to specify the name for the log group that a NOS/VE system is to provide the log recording function for. The default log group name for these commands is CATENET.

The command responses for the CDCNET logging commands have changed and are as follows.

DEFINE_SOURCE_LOG_GROUP (DEFSLG)

- . Source log group defined.
- . --WARNING-- Source log group defined, no message numbers specified.
- . --ERROR-- A source log group is already defined for the system.
- . --FATAL-- Not enough memory is currently available for required table space.

CHANGE_SOURCE_LOG_GROUP CHASLG

- . Source log group changed.
- . --WARNING-- No message numbers specified.
- . --ERROR-- Log group <name> not defined.

CANCEL_SOURCE_LOG_GROUP (CANSLG)

- . Source log group cancelled.
- . --WARNING-- Specified source log group cancelled. Source log group <name> was not defined.
- . --WARNING-- No source log groups defined.

DEFINE_RECORDER_LOG_GROUP DEFRLG

- . Recorder log group is defined for coupler node <xx>.
- . --WARNING-- Recorder log group is defined for coupler node <xx>. NP Interface for the coupler node is started but the logical link is down.
- . --WARNING-- Recorder log group is defined for coupler node <xx>. NP Interface for the coupler node is not started. Start NP Interface to enable log recording.
- . --ERROR-- A recorder log group is already defined for coupler node <xx>.
- . --ERROR-- NP Interface is not defined for coupler node <xx>.
- . --ERROR-- No default coupler node is defined. A coupler node must be specified.
- . --ERROR-- Recorder log groups cannot be defined for coupler node <xx>. Not enough memory is currently available for required table space.
- . --ERROR-- Recorder log groups cannot be defined for coupler node <xx>. Unable to initialize the log recording function.

CHANGE_RECORDER_LOG_GROUP (CHARLG)

- . Recorder log group is changed for coupler node <xx>.
- . --ERROR-- Recorder log group <name> is not defined for coupler node <xx>.
- . --ERROR-- No default coupler node is defined. A coupler node must be specified.

CANCEL_RECORDER_LOG_GROUP (CANRLG)

- . Recorder log group is cancelled for coupler node <xx>.
- . --WARNING-- Specified recorder log group is cancelled for coupler node <xx>. Recorder log group <name> was not defined.
- . --WARNING-- Recorder log groups were not defined for coupler node <xx>.
- . --WARNING-- Recorder log groups were not defined for the system.
- . --ERROR-- No default coupler node is defined. A coupler node must be specified.

12.8.5 Log Message Changes

1. The ACTION REQUIRED sections of some log messages in the Diagnostic Message Manual (60461600) should read as follows:
 - . ID 339: If condition persists or deteriorates, replace MPB board;
 - . ID 340: If condition persists or deteriorates, replace PMM board;
 - . ID 341 and 342: If condition persists or deteriorates, replace SMM board in indicated slot. If the board is being repaired, the error log information can be used to diagnose the problem.
2. Use the EXPCLM command to obtain the latest definition of each log message. Several messages were modified or added which are not included in the diagnostic manual.

12.8.6 Command Continuation

CDCNET's terminal interface does not support command line continuation

12.8.7 Accessing CDCNET Utilities - NOS/VE Only

To access any of the CDCNET utilities (ANACD, MANCC, NPA, ATTCF, REPCF, DEFCF, SETVL) on NOS/VE, the commands must be added to the users' command list. Enter the SCL command:

```
SET_COMMAND_LIST A=$SYSTEM.CDCNET.VERSION_INDEPENDENT.COMMAND_LIBRARY
```

before the CDCNET utility can be initiated.

Frequent user's of the CDCNET commands should add the SETCL command to their login prolog.

NOTE: This command is incorrectly listed for ANACD on page 2-2 of the Network Analysis Manual (60461590).

12.8.8 Retrieving a DI Dump File - NOS/VE Only

Page 2-4 of the Network Analysis Manual (60461590) incorrectly describes the format for a DI dump file name. The file names do not start with DUMP#FULL. Also, the example of a full path for the DI dump file should not conclude with a period (.).

12.8.9 Terminal Class for Teletype Model 43 - NOS & NOS/VE Dual State

The manuals incorrectly indicate that the Teletype Model 43 terminal should use terminal class 5; it is a terminal class 1 terminal. Attempts to use terminal class 5 produce unusable output. This also occurs if the `TERMINAL_MODEL` parameter is set for a model 43.

12.8.10 X.25 Configuration Recommendation - NOS Only

For X.25 DI to DI connection without a modem (which is configured by use of the `DEFINE_X25_TRUNK` directive), it is recommended that the `CLOCKING` parameter be set to `TRANSMIT`.

12.8.11 MDI Error Status - NOS Only

The error message 'MDI ERROR STATUS RECEIVED' is displayed when a problem is encountered with the MCI card in the DI. If continuous MDI ERROR STATUS RECEIVED messages appear in the NAM dayfile, the MCI board should be replaced.

12.8.12 Conversion of Transparent Attributes - NOS & NOS/VE Dual State

As recommended by Network Products when a NOS or NOS/VE dual-state application defines multiple terminal characteristics for transparency, the FNs 56, 57, 58, 59, 60, 69, 70, and 146 should be sent together to ensure proper results. The conversion of these FNs to CDCNET attributes is quite complex. If an application is having problems with transparent input, the following should be reviewed very carefully. There are some subtle differences between the CCP and CDCNET Transparent Input features.

1. TRANSPARENT FORWARD/TERMINATE ON MESSAGE LENGTH

If FNs 57 and 58 are zero or not specified, TLM is set to NONE.

If FNs 57 or 58 are specified, TML is set to the value of FN 58 plus $256 * (\text{value of FN 57})$. If FN 70 is not specified or has a value of 0, then TLM is set to TERMINATE. If FN 70 has a value of 1 and TML is 1, then TLM is set to FORWARD. If FN 70 has a value of 1 and TML is not 1, then TLM is set to FORWARD_EXACT.

NOTE - For multi-message transparent input, TLM is set to FORWARD if TML is 1, and TLM is set to FORWARD_EXACT if TML is greater than 1. CCP is always FORWARD_EXACT.

2. TRANSPARENT FORWARD/TERMINATE ON CHARACTER

If FN 56 is not specified or has a value of 0, then TCM is set to NONE.

If FN 56 has a value of 1 and FN 70 is not specified or has a value of 0, then TCM is set to TERMINATE.

If FN 56 has a value of 1 and FN 70 is specified with a value of 1, then TCM is set to FORWARD_TERMINATE.

If FN 59 is specified, then TFC is set to the value of FN 59.

If FN 59 is specified as CR and the terminal's parity had been previously set to IGNORE, TFC is set to (OD(16),8D(16)).

If FN 69 is specified, then TTC is set to the value of FN 69.

If FN 69 is specified as CR and the terminal's parity had been previously set to IGNORE, TTC is set to (OD(16),8D(16)).

If FN 70 has a value of 1 and 59 has been specified but 69 hasn't, TTC will be set equal to TFC.

3. TRANSPARENT FORWARD/TERMINATE ON TIMEOUT

If FN 60 is not specified or has a value of 0, then TTM is set to NONE.

If FN 60 has a value of 1 and if either FN 70 or 146 was not specified or either has a value of 0, then TTM is set to TERMINATE.

If FN 60 has a value of 1 and if both FNs 70 and 146 were also specified with a value of 1, then TTM is set to FORWARD.

12.8.13 Effects of Changing Terminal Classes - NOS & NOS/VE Dual State

Changing the Terminal Class affects many of the CDCNET attributes. The following attribute settings are made for all Terminal Class changes:

BA=0, CFC=TRUE, CLC=CAN, ELC=CR, ELP=LFS, EPC=LF, EPP=CRS, HP=NO, NCC='%', P=EVEN, PCF=FALSE, SA=SEND, SBC=FALSE, SND=FALSE, TCM=TERMINATE, TFC=CR, TLM=TERMINATE, TML=2043, TTM=NONE.

In addition to the above global effects, the following chart indicates the changes made when the specified Terminal Class change is selected.

	TERMINAL CLASS						
	1	2	3	5	6	7	8
BC	BS	BS	BS	none	BS	BS	BS
CRD*	2	0	0	1	0	0	0
LFD*	1	0	0	3	3	0	0
PL	0	24	30	24	27	24	35
PW	72	80	80	80	74	80	74
FL	yes	no	no	no	no	yes	no
FFS	6 LFs	EM/ CAN	FF	ESC R	FS	ESC [H ESC [J	ESC FF
CRS	CR	CR	CR	ESC G	CR	CR	CR
LFS	LF	LF	LF	ESC B	0	LF	LF
E	no	no	no	no	no	yes	no
FFD	Not changed for TC 1..7						999ms

* - Millisecond value is dependent on linespeed

12.8.14 CDCNET Mapping of NAM FN/FV Pairs - NOS & NOS/VE Dual State

CDCNET terminal attributes can be changed using the IAF 0016 control byte interface or the NAM CTRL/CHAR supervisory message interface.

Field Number	Field Name	Field Value	CDCNET changes
25	BF	0 1 2	PCF=TRUE PCF=FALSE, IBS=100 PCF=FALSE, IBS=200
30	XBZ1	all	none
31	XBZ2	all	none
32	LK	0 1	SA=SEND SA=DISCARD
33	HD	all	none
34	TC	all	Note 1
35	PW	n	PW=n
36	PL	n	PW=n
37	PG	0 1	HP=FALSE HP=TRUE
38	CN	char	CLC=char
39	BS	char	BC=char
40	CT	char	NCC=char
41	AB	all	none
42	B1	all	none
43	B2	all	none
44	CI	n	CRD=n*linespeed factor
45	LI	n	LFD=n*linespeed factor
46	CA	all	none
47	LA	all	none

Field Number	Field Name	Field Value	CDCNET changes
49	EP	0 1	E=FALSE E=TRUE
50	PA	0 1 2 3 4	P=ZERO P=ODD P=EVEN P=NONE P=NONE
51	BR	0 1	BKA=0 BKA=1
52	XPT	0 1	IEM=NORMAL IEM=TRANSPARENT
53	IN	all	none
54	OP	0 1 2	FL=FALSE FL=TRUE none
55	FA	0 1	EPC=LF, SBC=FALSE, SND=FALSE EPC=NUL, SBC=TRUE, SND=TRUE
61	EL	char	ELC=char
62	ELO	all	none
63	CPEL	0 1 2 3	ELP=NONE ELP=CRS ELP=LFS ELP=CRSLFS
64	EB	all	none
65	EBO	all	none
66	CPEB	all	none
67	IC	0 1	CFC=FALSE CFC=TRUE
68	OC	0 1	CFC=FALSE CFC=TRUE

Field Number	Field Name	Field Value	CDCNET changes
71	CP	0	ELP=NONE, EPP=NONE
		1	ELP=LFS, EPP=CRS
87	FDLX	0	IOM=SOLICITED
		1	IOM=FULLDUPLEX
102	PP	char	EOS=char
112	NTA	0	IOM=UNSOLICITED
		1	IOM=SOLICITED
147	CRI	n	CRD=n*4
148	LFI	n	LFD=n*4
149		0	none
		1	none

Note 1: Also see section on Effects of Changing Terminal Class.

12.8.15 Effects of NOS TRMDEF Command - NOS and NOS/VE Dual State Only

CDCNET terminal attributes can be changed by a terminal user executing a NOS TRMDEF command. The following table shows the mapping between TRMDEF parameters and the corresponding CDCNET attributes.

TRMDEF PARAM	CDCNET ATTRIBUTE CHANGES
AB	No changes
BF=0 BF=1 BF=2	PCF=TRUE PCF=FALSE, IBS=100 PCF=FALSE, IBS=200
BS=char	BC=char
B1	No changes
B2	No changes
CI=n	CRD=n*linespeed factor
CN=char	CLC=char
CP=Y CP=N	ELP=LFS, EPP=CRS ELP=NONE, EPP=NONE
CT=char	NCC=char
DL=Xxx DL=Cn DL=TO	TCM=TERMINATE, TFC=xx, TTC=xx TLM=TERMINATE, TML=n TTM=TERMINATE
EB	No changes
EL=char EL=EB EL=EL EL=CR EL=LF EL=CL EL=NO	ELC=char No changes No changes ELP=CRS ELP=LFS ELP=CRSLFS ELP=NONE
EP=Y EP=N	E=TRUE E=FALSE
FA=Y FA=N	EPC=NUL, SBC=TRUE, SND=TRUE EPC=LF, SBC=FALSE, SND=FALSE
IC=YES IC=NO	CFC=TRUE CFC=FALSE

TRMDEF PARAM	CDCNET ATTRIBUTE CHANGES
IN=BK IN=KB IN=PT IN=X IN=XK IN=XP	No changes IEM=NORMAL IEM=NORMAL IEM=TRANSPARENT IEM=TRANSPARENT IEM=TRANSPARENT
LI=n	LFD=n*linespeed factor
LK=Y LK=N	SA=DISCARD SA=SEND
OC=YES OC=NO	CFC=TRUE CFC=FALSE
OP=DI OP=PR OP=PT	FL=FALSE FL=TRUE No changes
PA=E PA=N PA=O PA=Z PA=I	P=EVEN P=NONE P=ODD P=ZERO P=NONE
PG=Y PG=N	HP=TRUE HP=FALSE
PL=n	PL=n
PW=n	PW=n
SE=Y SE=N	SBC=TRUE, EPC=NUL SBC=FALSE, EPC=LF
XL=Xxx XL=C1 XL=Cn XL=TO	TCM=FORWARD_TERMINATE, TFC=xx, TTC=xx TLM=FORWARD, TML=1 TLM=FORWARD_EXACT, TML=n TTM=TERMINATE

NOTE

TRMDEF commands containing DL and/or XL that do not specify all of the transparent delimiting conditions, i.e. Xxx, Cn, and TO, will cause the CDCNET transparent mode attributes, corresponding to the not selected delimiting conditions, to be set to a value of NONE. As an example:

TRMDEF,XL=C1 causes TLM and TTM to be set to NONE because Xxx and TO were not specified.

12.9 Future NOS Considerations - NOS Only

In a future release, terminal user procedures, terminal definition procedures and DI configuration procedures will be combined on to their own respective libraries, similar to what is done on NOS/VE for CDCNET1.1. Several practices you can use now to ease migration to the new format are listed below:

- a. Maintain all terminal user procedures, terminal definition procedures and DI configuration files on the same user name.
- b. Include PROC and PROCEND statements in DI configuration files. The proper statements are:

PROC SYSTEM_080025ssssss as the first line of the file

PROCEND SYSTEM_080025ssssss as the last line.

p 1=1

(ssssss are the last six digits of the system identifier.)

Chapter 13 COBOL

13.1 Significant Problems

The following items are significant problems in the area of COBOL:

1. Compilation Hang Using COPY Statement
2. Duplicate User Defined Names Not Diagnosed
3. COBOL Parameter ERROR Does Not Work As Documented

13.1.1 Compilation Hang Using COPY Statement

Users must observe the following rules for use of COPY statements.

1. A COPY statement must end with a period and only one COPY is allowed* in each sentence.
2. A nested COPY using the COPY statement in the deck being copied is not allowed.

Deviation from these rules may cause compilations to hang instead of issuing the proper syntax errors.

PSR CB80029, CB8A775

13.1.2 Duplicate User Defined Names Not Diagnosed

Duplicate user defined names will not be diagnosed at compilation time as they were with previous level systems. Currently the program will abort at runtime if the name is referenced.

PSR CB8A699

13.1.3 COBOL Parameter ERROR Does Not Work As Documented

The VE COBOL Users Guide specifies that if the ERROR parameter is omitted and the LIST parameter is specified, diagnostics will be listed on the file specified by the LIST parameter. NOS/VE 1.2.1 L670 COBOL incorrectly writes the diagnostics to the \$ERRORS file.

PSR CB80039

13.2 Notes and Cautions

The following items are notes and cautions in the area of COBOL:

1. All COBOL Programs Must Be Recompiled
2. AUDIT Compilation Parameter
3. Interprogram Communication
4. New COBOL Reserved Words
5. COBOL Doesn't Support Multiple Statement Debugging
6. COBOL Parameter BASE_LANGUAGE=ANS85 Has No Effect

13.2.1 All COBOL Programs Must Be Recompiled

Because of operating system changes, all COBOL programs must be recompiled.

13.2.2 AUDIT Compilation Parameter

The AUDIT parameter allows compilation of an ANS standard COBOL program for Federal Compiler Testing Center audit testing. At NOS/VE 1.2.1 L670, the AUDIT parameter enables audit testing of 1974 ANS Programming Language COBOL programs. With NOS/VE 1.2.2, the AUDIT parameter will change to enable audit testing of 1985 ANS Programming Language COBOL programs. CDC extension keywords will be diagnosed as illegal if AUDIT=TRUE is specified.

13.2.3 Interprogram Communication

At NOS/VE 1.2.1 L670, runtime fields used for storing and passing file names specified in the ASSIGN clause have been changed from a 31 character string to a 512 character string in preparation for future enhancements. When passing a file as a parameter from a COBOL program to a non-COBOL subroutine with the intent of using the file name in an AMP\$ interface, it is necessary to assign the name to a 31 character string variable and pass that string rather than the original to the AMP\$ interface. The deck CBD\$FILE_NAME_PARAMETER from the source library file \$\$SYSTEM.COMMON.PSF\$EXTERNAL_INTERFACE_SOURCE includes this change to the size of the file name field.

COBOL runtime routines by default perform optimizations whenever possible to improve the performance of I/O for sequential files. For programs which pass open sequential files as parameters to a subroutine which performs AMP\$ interface operations on the files, these optimizations produce unpredictable results. When using files in the above manner, it is now necessary for the user to tell the COBOL runtime routines not to do the optimizations. This is

done by defining the SCL variable CBV\$IO_OPT_LOW as follows before execution of the program:

```
CREATE_VARIABLE cbv$io_opt_low KIND=BOOLEAN VALUE=TRUE
```

13.2.4 New COBOL Reserved Words

The following is a list of words added to the reserved word list of the 1985 American National Standard Programming Language COBOL. Use of these words in a manner different from that specified in the VE COBOL Users Guide will result in diagnostics. Users of NOS/VE 1.2.1 L670 COBOL may suppress these diagnostics if they specify AUDIT=TRUE and their program is a 1974 ANS COBOL program. This tells the NOS/VE 1.2.1 L670 COBOL compiler to recognize as reserved words only those words defined as reserved words in the 1974 ANSI Standard.

With the NOS/VE 1.2.2 release, AUDIT=TRUE will no longer perform this function because it will be used to test for 1985 ANS COBOL. With NOS/VE 1.2.2, the user will be able to change the list of words recognized as reserved by specifying the BASE_LANGUAGE (BL) parameter. BL=COBOL5 will cause the COBOL5 reserved word list to be used. BL=ANS74 will cause the NOS/VE 1.2.1 L670 COBOL reserved word list to be used.

alphabet	end-divide	external
alphabetic-lower	end-evaluate	false
alphabetic-upper	end-if	global
alphanumeric	end-multiply	initialize
alphanumeric-edited	end-perform	numeric-edited
any	end-read	order
binary	end-receive	other
class	end-return	packed-decimal
common	end-rewrite	padding
content	end-search	purge
continue	end-start	reference
day-of-week	end-subtract	standard-2
end-add	end-unstring	test
end-call	end-write	then
end-compute	evaluate	true
end-delete		

13.2.5 COBOL Doesn't Support Multiple Statement Debugging

The STATEMENT parameter of the DEBUG SET_BREAK command (a new feature in DEBUG) is not supported by COBOL. The user is unable to specify break definitions on particular statements in multiple-statement lines. Previous releases of DEBUG recognized only the first statement in the multiple-statement line; COBOL support continues with this level of statement recognition for NOS/VE 1.2.1 L670.

13.2.6 COBOL Parameter BASE_LANGUAGE=ANS85 Has No Effect

Specification of the BASE_LANGUAGE=ANS85 parameter on the COBOL command has no effect on the compilation process. The major effect of this is that specification of the parameter FIPS=OBSOLETE will result in termination of the compilation during cracking of the parameter list.

Chapter 14 Configuration Management

14.1 Significant Problems

The following items are significant problems in the configuration management area:

1. EDIT_PHYSICAL_CONFIGURATION Command May Fail
2. NOS - NOS/VE Dual-State PP Allocation
3. All Controllers OFF Requires Units OFF

14.1.1 EDIT_PHYSICAL_CONFIGURATION Command May Fail

At NOS/VE 1.2.1 L670, if during execution of the PCU subcommands in the configuration prolog an error is encountered, the system will give a PCU prompt instead of a PCE prompt. Any attempt to use the EDIT_PHYSICAL_CONFIGURATION command will result in the following error message:

```
--ERROR CM 779-- EDIT_PHYSICAL_CONFIGURATION cannot be executed within
EDIPC session.
```

To correct the problem, type QUIT to reenter the PHYSICAL_CONFIGURATION_UTILITY.

PSR NVOJ025

14.1.2 NOS - NOS/VE Dual-State PP Allocation

When running in a dual-state environment with NOS, there are some instances when NOS/VE will not be able to obtain additional PP's from NOS. This behavior is partly explained based on how NOS allocates PP's to NOS/VE and on the type of PP's NOS/VE is presently requesting.

The following rules are used by NOS in allocating PP's to NOS/VE:

1. NOS must have a minimum of 4 pool PP's available for its own use. The 4 pool PP's do not include dedicated PP's which contain MTR, DSD, SCD and similar programs. If the current request would cause the minimum to fall below 4, the request will not be granted.

2. If the mainframe is an 810 or 830 with 20 PP's, then an additional requirement is that there must be at least 2 driver PP's available for use by NOS. (Driver PP's are numbered 20 - 31). Again, dedicated PP's are excluded from this count.

The minimum number of pool PP's, and the minimum number of driver PP's are actually assembly constants in the NOS common deck COMSVED. The minimum number of pool PP's is named MINP, and the minimum number of driver PP's is named MINDP. The released values of these parameters, 4 for MINP and 2 for MINDP, are also the minimum values to which these parameters should be set.

At the present time on an 810 or 830 with 20 PP's, NOS/VE requests all PP's as driver PP's.

On a CYBER 810 or 830 with 20 PP's in dual state mode, the maximum number of PP's NOS/VE can use for drivers is 7. This is because 2 of the driver PP's are reserved for NOS, and 1 PP contains SCD.

PSR NVOJ250

14.1.3 All Controllers OFF Requires Units OFF

In the physical configuration, if all controllers connected to mass storage units have their state set to OFF, then those mass storage units must also have their state set to OFF.

PSR NVOJ515

14.2 Notes and Cautions

The following items are notes and cautions in the configuration management area:

1. Memory Configurations up to 128 MB
2. Support of 7154 Controller
3. 7165-2x/895 or 887-1 Requires a Minimum 4K Page Size
4. NOS/VE Minimum Hardware Requirements in Dual-State Mode
5. State Change of MDI/MTI
6. FCA Levels

14.2.1 Memory Configurations up to 128 MB

NOS/VE supports memory configurations of up to 128 MB. Up to 16 MB can be used by NOS or NOS/BE with the remainder being used by NOS/VE.

Memory dumps are created on magnetic tape at the time of a system failure to assist in the analysis of the failure. For this purpose, configurations with a central memory size up to 32 MB must include a nine track tape unit with 1600 PE or 6250 GCR recording capabilities. Configurations with a central memory size greater than 32 MB must include a nine track tape unit with 6250 GCR recording capabilities. Configurations with central memory size greater than 64 MB can use two separate reels of magnetic tape for system dumps. Also, temporary mass storage space on both NOS/VE and NOS (256 PRUs/MB) or NOS/BE equal to the memory size must be provided to dump and analyze these tapes.

14.2.2 Support of 7154 Controller

The following information is provided for those sites installing NOS/VE on a configuration using 7154 controllers. Users should be aware that NOS/VE support of 7154 controllers is planned to be discontinued in the future.

To use a 7154 controller in a NOS/VE system, make the following changes:

1. Use \$7154_1 as the Product_Identification in Configuration Prolog files.
2. Use \$7154_1 when defining the deadstart controller.
3. Enter OTHER when at the deadstart boot display menu for disk.

7154 disk subsystem failure data are reported as statistic CM4100 in the engineering log.

14.2.3 7165-2x/895 or 887-1 Requires a Minimum 4K Page Size

At NOS/VE 1.2.1 L670, a page size of 4K or greater is required if 7165-2x/895 or 887-1 disks are to be used at your site.

14.2.4 NOS/VE Minimum Hardware Requirements in Dual-State Mode

The following minimum hardware is required to support NOS/VE in the dual-state environment (this information is basically the same information that is found in Volume II of the Corporate Pricing Manual).

- a. 8 MB of memory (this assumes no activity on NOS or NOS/BE except for those functions necessary to support NOS/VE such as interactive support, print output, etc.). This is a minimum memory requirement for any dual-state configuration.
- b. 15 PP's as a minimum on all machines with the exception of the 810/830 which require a 20 PP configuration.

- c. Disk controller channel; one dedicated as a minimum for NOS/VE and one dedicated to NOS or NOS/BE as a minimum. If you use a 7155 controller and an 885 disk unit, the controller must be a model "B" (or a model "A" with option 65290-1 installed - this option is indicated by tag number FV715-A) for large sector formatting.
- d. Disk storage capacity; 350 MB as a minimum for NOS/VE and one physical disk unit for NOS or NOS/BE. NOS/BE dual-state configurations do not currently support the 834, 836 or 895 disks and they should not be included in a NOS/BE dual-state configuration.
- e. Tape controller and tape unit; at least one tape controller and tape unit required per machine. The controller/unit can be shared between NOS or NOS/BE and NOS/VE. Two tape units are required if you are going to use labeled tapes.
- f. In most cases, additional channels are required and are either added as part of the first PP increment on the 840 machines and up, or are required before a PP upgrade can be added to the 810/830 machines.
- g. In all configurations the CC634B console is required as the NOS/VE operator interface. Refer to Installation Note and Caution item "CC634B Console" for additional information.
- h. If files are to be printed out at the central site, the site should have a printer with ASCII capability.

The minimum hardware will let you install and initialize NOS/VE and accomodate a minimum amount of usage. Additional memory and disk controllers/units may be necessary to support any activity on NOS or NOS/BE or increased activity on NOS/VE.

14.2.5 State Change of MDI/MTI

The LCU command `CHANGE_ELEMENT_STATE` cannot be used to change the state of MDI and MTI devices. In order to change the state of these devices, one must do it at PCU time during deadstart, using the `CHANGE_ELEMENT_DEFINITION` in the `EDIT_PHYSICAL_CONFIGURATION` subutility.

14.2.6 FCA Levels

FCA levels are described in the Configuration Management section of the Software Availability Bulletin (SAB).

Chapter 15
CYBIL

15.1 Significant Problems

The following items are significant problems in the area of CYBIL:

1. INLINE Functions
2. FOR Loop
3. Alignment Inheritance
4. Alignment Ignored for ALLOCATE
5. Small Fields As Reference Parameters
6. Handling Large Constants

15.1.1 INLINE Functions

Incorrect code is generated when INLINE functions are referenced in WHILE statements or in logical expressions. For example, if fcn is given the INLINE attribute, incorrect code is generated for the following kinds of statements:

1. For

```
WHILE fcn (arguments) DO
```

the function is not reevaluated at the top of the loop. The generated code is as if the following source code were written:

```
b := fcn (arguments);  
WHILE b DO
```

2. For

```
IF (ptr <> NIL) AND fcn (ptr) THEN
```

the function is evaluated before the IF statement. The generated code is as if the following source code were written:

```
b := fcn (ptr);  
IF (ptr <> NIL) AND b THEN
```

Because of the above, a number of problems can occur. For example, fcn may not be prepared to handle a NIL pointer.

3. For

```
b1 := (y <> 0) AND fcn (x, y);
```

the function is evaluated before the statement. The generated code is as if the following source code were written:

```
b2 := fcn (x, y);  
b1 := (y <> 0) AND b2;
```

Again, problems can occur in fcn; e.g., suppose fcn divides x by y.

For the WHILE statement problem, there is no workaround other than to not use the INLINE attribute. For INLINE functions in logical expressions, there is no workaround other than moving the function references to simple assignment statements as illustrated above.

PSR CILA926, CILA929, CILB036, CILB082, CILB160

15.1.2 FOR Loop

At OPT=HIGH, incorrect register allocation is sometimes done for the branch instruction at the end of a FOR loop. Only one register is referenced in the branch instruction at the bottom of the loop, for example,

```
BRINC    X6,X6,label
```

Depending on the instruction at the bottom of the loop, either an infinite loop results or the loop is only executed once.

The problem seems to occur if the terminal value for the loop resides in static data and both the loop index and the terminal value are defined prior to the loop. For example,

```
VAR  
    index: integer,  
    terminal_value: [STATIC] integer;  
  
index := 31;  
terminal_value := index;  
FOR index := 1 TO terminal_value DO  
    etc;  
FOREND;
```

A number of workarounds are possible, such as using a separate automatic variable for the loop index, or using an automatic variable for the terminal value.

PSR CILB243, CILB252

15.1.3 Alignment Inheritance

If ALIGNED is used on a field other than the first field, the record does not inherit the alignment of this field. The ALIGNED field will be mapped incorrectly. For example,

```
TYPE
  r1_f1: boolean,
  r1_f2: r2_type,
  RECEND,
  r2_type = packed RECORD
    r2_f1: boolean,
    r2_f2: ALIGNED [0 MOD 8] 0..0ffff(16),
  RECEND;
```

For the above, a workaround would be to make r2_f1 ALIGNED, also.

PSR CILA742, CILA746, CILA747, CILB012, CILA247

15.1.4 Alignment Ignored for ALLOCATE

The alignment specification for a type is ignored when a variable defined as a pointer to that type is used in an ALLOCATE statement.

In the following example, the address returned by the ALLOCATE is not necessarily 0 MOD 8:

```
TYPE
  rec = RECORD
    first: ALIGNED [0 MOD 8] INTEGER,
    .
    .
    .
  RECEND;
VAR
  rec_ptr: ↑rec;
ALLOCATE recptr;
```

PSR CILB063

15.1.5 Small Fields As Reference Parameters

A record field which is less than one word in length passed as a reference parameter is not handled correctly.

In the following example, the assignment to variable r in procedure p1 causes an overwrite of y.ch due to this problem.

```
MODULE m;  
  TYPE  
    word = -8000(16) .. 7fff(16);  
  
  PROCEDURE p1 (VAR r: word);  
    r := 0;  
  PROCEND p1;  
  
  PROCEDURE p2;  
    VAR  
      y: packed RECORD  
        w: ALIGNED word,  
        ch: char,  
      RECEND;  
  
    p1 (y.w);  
  PROCEND p2;  
MODEND m;
```

PSR CILB043

15.1.6 Handling Large Constants

Constants with values greater than 60 bits are not given correct values by the compiler. In particular, constants used for static initialization cannot exceed 60 bits.

PSR CIL0057, CILB184, CILB239

Chapter 16 DVS

16.1 Significant Problems

The following items are significant problems in DVS:

1. General DVS Problems
2. 835 System DVS Problem
3. 840 through 860 System DVS Problems
4. 990 System DVS Problem

16.1.1 General DVS Problems

1. It is strongly recommended that the REDUCE_TASK_PRIORITY parameter be used when submitting DVS batch jobs. Reducing task priority automatically reduces the priority of all tasks in the job as well as the DVS job itself. Because the job's priority was reduced, DVS receives the CPU less often the more active the system becomes. Thus it may take a noticeably longer time to terminate than is expected.
2. If REDUCE_TASK_PRIORITY is used when executing DVS tests interactively, the priority of the entire job is reduced to the lowest priority valid for the job class.

To display the priority of the current job, enter:

```
DISPLAY_JOB_ATTRIBUTES DO=DISPATCHING_PRIORITY
```

At any time, the priority of the job may be changed by using the following command:

```
CHANGE_JOB_ATTRIBUTES DP=new-dispatching-priority
```

where "new-dispatching-priority" is the new priority of the job.

3. DVS Version 2.1 includes a Communications Confidence Test (COMCT) which verifies a communications path to a CDCNET system via the NAM/VE XNS Internet Services User Ring Interface.

4. When running with User Mask clear (via SET_MODE user_mask_selection=false) in RFST and DEBUG, BDP invalid data results are undefined, and on some machines, not repeatable. On these machines, when the user mask is cleared, all BDP op codes should be disabled using the following sequence:

```
ADD_INSTRUCTIONS bdp_op_codes=true
SET_MODE instruction_list_selection=deselect
```

When running with User Mask clear in DEBUG, BDP op codes, Single-Precision Floating-Point op codes and Double-Precision Floating-Point Quotient (op code 37) should also be disabled using the following command:

```
ADD_INSTRUCTIONS op_code=(30(16),31(16),32(16),33(16),3a(16) ..
,3b(16),3c(16),98(16),99(16),9a(16),9b(16),9e(16),37(16))
```

16.1.2 835 System DVS Problem

Selection of diagnostics BIMM, NUMR, RFST, and DEBUG is invalid for the NOS/VE 1.2.1 L670 release.

16.1.3 840 through 860 System DVS Problems

1. Diagnostic RFST detects known BDP end-case failures. When using the RUN_DVS command, do not alter the RFST_DELETE_BDP parameter from its default setting. It is IMPERATIVE that the parameter remain at its default value until the FCO 46111 is installed.
2. Selection of the DEBUG diagnostic is invalid for the NOS/VE 1.2.1 L670 release.

16.1.4 990 System DVS Problem

Selection of the DEBUG diagnostic is invalid for the NOS/VE 1.2.1 L670 release.

Chapter 17
File Management Utility (FMU)

17.1 Significant Problems

The following items are significant problems in the area of FMU:

1. Maximum Block Length for IBM Tape Files

17.1.1 Maximum Block Length for IBM Tape Files

The maximum block length allowed for migrating IBM tape files is 4128 bytes. However IBM adds a 4-byte control word in addition to the specified block length to make it 4132 bytes which will cause migration to fail. Users should specify at most 4124 bytes when generating the IBM data file for migration.

PSR FM8A247

Chapter 18

FORTTRAN/VE 1

18.1 Significant Problems

The following items are significant problems in the area of FORTTRAN/VE 1:

1. Enforcing Flushing of Terminal Output

18.1.1 Enforcing Flushing of Terminal Output

Terminal output is not automatically flushed to the terminal when it is written. Doing so automatically would have a significant performance impact in certain circumstances. This means that output that has been written to the terminal does not actually show up at the terminal until either

1. a large (several thousand bytes) buffer is filled,
2. input is requested from the terminal,
3. the FORTTRAN ENDFILE statement is executed, or
4. the FORTTRAN program completes execution.

Programs that produce terminal output without satisfying any of the last three items above may be subject to a substantial time lag between the point in time when the FORTTRAN program produces output and the point in time when it shows up at the terminal. To alleviate this problem, a temporary solution has been provided.

If the job is interactive and the user has declared an SCL integer variable called `OSV$FLUSH_KLUDGE`, then flushing of all terminal files will be forced for list-directed, formatted, and namelist output. The existence of this variable is checked for only during program initialization. This is remembered and will hold throughout the entire execution of the program. If the variable is created or deleted dynamically during the program's execution, it will not be detected. If the variable is to be recognized by a FORTTRAN program in an SCL procedure or from within an embedded procedure, the variable should have an appropriate `JOB`, `XDCL` or `XREF` scope for the situation. If `OSV$FLUSH_KLUDGE` does not exist, terminal output files are not explicitly flushed. The ability to force flushing of terminal output for a particular file is still retained via the `ENDFILE` statement and is independent of the existence of an `OSV$FLUSH_KLUDGE` variable.

PSR NV0I942

18.2 Manual Errata

The following items are Manual Errata in the area of FORTRAN/VE 1:

1. Ignore References to FORTRAN/VE 2

18.2.1 Ignore References to FORTRAN/VE 2

References in the manuals to FORTRAN/VE 2 should be ignored. FORTRAN/VE 2 is not yet a released product.

Chapter 19 HPA/VE

19.1 Significant Problems

The following items are significant problems in HPA/VE:

1. HPA/VE Reloads Report's OPEN/CLOSED Status When It Shouldn't
2. HPA/VE Does Not Recognize MAP 5 Engineering Log Error Entries

19.1.1 HPA/VE Reloads Report's OPEN/CLOSED Status When It Shouldn't

You cannot reload an archived HPA/VE report from an HPA/VE archive tape with the report still existing in the Maintenance Action Log database. This is normal operation. However, the OPEN/CLOSED status of the report in the database INDEX OF REPORTS gets overwritten with the OPEN/CLOSED status of the report from the HPA/VE archive tape. This is not normal. The existing report in the database is not affected.

PSR HP8A061

19.1.2 HPA/VE Does Not Recognize MAP 5 Engineering Log Error Entries

NOS/VE is not logging a configuration entry for the MAP 5 element when it is included in the NOS/VE system configuration. In addition, the MAP 5 error entry's descriptive data string is incorrect. Because there is no MAP 5 configuration entries, HPA/VE does not recognize MAP 5 error entries when it finds such entries in the Engineering Log. HPA/VE will log MAP 5 error entries in its UNKNOWN circular error log and produce both an HPA_VE error incident report and an UNKNOWN error incident report for each MAP 5 error entry it processes.

The HPA_VE error incident report will contain the error message

```
CONVERSION FROM LOGICAL TO PHYSICAL NOT ACCOMPLISHED FOR ELEMENT  
NAME = xxx      , DEGRADABLE_FUNCTION_SPECIFIED = TRUE
```

where "xxx" is the AP 5 element name assigned by the site.

The UNKNOWN error incident report will contain the MAP 5 error statistic (logging code CM0006100), its descriptive data string and its counters. For a definition of the various MAP 5 error entry counters, refer to the Configuration Management Statistics section, statistic CM6100, in the NOS/VE

System Performance and Maintenance Manual (60463915).

PSR NVOJ889, NVOJ890

19.2 Notes and Cautions

The following items are notes and cautions when using HPA/VE:

1. Delete Catalogs before Installing NOS/VE 1.2.1 the First Time
2. Running ACCESS_HPA from the System Console
3. Mainframe Errors Logged in NOS or NOS/BE BML

19.2.1 Delete Catalogs before Installing NOS/VE 1.2.1 the First Time

If your site has been running a pre-released version of NOS/VE 1.2.1, L664, you should delete all of the \$SYSTEM.HARDWARE_MAINTENANCE.HPA.DATA catalog files and all of the \$SYSTEM.HARDWARE_MAINTENANCE.HPA.DATA.HPF\$CIRCULAR_ERROR_LOGS catalog files prior to installing NOS/VE 1.2.1 L670. The released version of HPA/VE (01.00) uses file formats which are not compatible with early prerelease versions of HPA/VE. RUN TIME ERRORS will occur if the old formatted HPA/VE files are accessed by the new released HPA/VE.

DO NOT DELETE the above named files if your current running version of HPA/VE is 01.00 or greater. The first line of the HPA/VE MAIN MENU display contains the HPA/VE version number ("HPA/VE 01.00")

19.2.2 Running ACCESS_HPA from the System Console

When running ACCESS_HPA from the System Console, the user does not have to end ACCESS_HPA to process operator requests. Pressing the STOP key will put the user into PAUSE MODE, at which time operator commands can be entered. Entering "RESUME_COMMAND" or "RESC" will put the user back into ACCESS_HPA where he was when he paused.

19.2.3 Mainframe Errors Logged in NOS or NOS/BE BML

On dual state systems, fatal mainframe errors that down the system and mainframe errors that occur during the dead start process are logged in the NOS or NOS/BE BML, and are not available to HPA/VE for processing. The CE must analyze mainframe error information from both the NOS or NOS/BE HPA reports and from the NOS/VE HPA/VE reports in order to have a total picture of his mainframe's errors.

In addition, DFT's counts of total mainframe errors logged (to both NOS and NOS/VE logs, or to both NOS/BE and NOS/VE logs) are logged in the NOS/VE Engineering Log. This results in HPA/VE's mainframe elements' SQC statistics indicating larger corrected and uncorrected error counts than the actual number of error entries analyzed. The CE will not be able to correlate the number of detailed error entries analyzed by HPA/VE with the total number of error occurrences reported by HPA/VE.

Chapter 20 IM/DM

20.1 Installation

IM/DM is the Information Management/Data Management system. This section discusses:

1. IM/DM installation (IM/DM was not previously installed on your system)
2. IM/DM upgrade (IM/DM is already installed on your system)
3. Initiation of the DM kernel

For performance reasons, Control Data strongly recommends that IM/DM be executed with a central memory of at least 16 megabytes, although it can be executed with a central memory of 8 megabytes.

Also, IM/DM is designed to use a mainframe page size of 4K bytes for NOS/VE. Use of any other page size may have a negative impact on performance.

20.1.1 IM/DM Initial Installation

1. Use the ADMINISTER_USER command of the Family Administration Utility to create the user name DMROOT (with your own password) in family create in step 1. The DMROOT user name and all IM/DM users must belong this family. When you validate DMROOT, include the Family Administration Utility subcommand:

```
set_minimum_ring 4
```

DMROOT is where the IM/DM files reside and also where the DM kernel executes.

2. Use the Upgrade Software Utility to install the IM/DM products.

NOTE

Each INSTALL_PRODUCT subcommand initiates a NOS/VE batch job that requests a tape, loads files from the tape, rewinds the tape, and processes the files. You must wait for each batch job to complete before entering the next INSTALL_PRODUCT command (please note that the batch job may still be executing, even

though the tape has been unloaded). You may check on the status of each job using the VED AJ display.

The IM_DM_BASE_I and IM_DM_BASE_II jobs process a large number of files and, therefore, require considerably longer processing time than the remaining jobs.

Use the UPGRADE SOFTWARE subcommand, DEFINE_IM_DM_FAMILY, to define the family that IM/DM will reside in.

From the system console, enter the following commands (but only for the IM/DM tapes that you have):

```
upgrade_software
define_im_dm_family family-name
install_product im_dm_base_i
install_product im_dm_base_ii
install_product im_dm_query
install_product im_dm_report
install_product im_dm_processor
install_product im_dm_dli
quit
```

??where "family-name" is??

3. Run the IM/DM installation procedure. To do this, log in under the user name DMROOT, and enter the command:

```
$user.relc3.install
```

- a. Verifies DMROOT. To execute this procedure file, you must be logged in to DMROOT. No user response is required for this step.
- b. Removes previous DM commands. This step removes the DM environment and commands from the job library and command lists. No user response is required for this step.
- c. Creates catalog permits. These permits are created for \$USER catalog:

```
G=USER AM=(ALL,CYCLE,CONTROL) SM=NONE
G=FAMILY AM=(READ,EXECUTE) SM=(READ,EXECUTE)
G=PUBLIC AM=(READ,EXECUTE) SM=(READ,EXECUTE)
```

You must answer YES to this prompt.

- d. Creates file permits. Ring attributes and file permits for the CYBIL support and command libraries are set to their required values. It also creates the file permissions necessary for all users to access the CYBIL support and command libraries. You must answer YES to this prompt.

- e. Sets the family name, user name, and password for the NOS/VE account (DMROOT) which will contain the IM/DM binaries. This step first prompts to see if you would like to change these values. Answer YES to this prompt. Then enter the name of the family that will contain the IM/DM binaries, the name DMROOT for the user name, and the name of the password for DMROOT. This information will be used to edit the login information in the jobs which are submitted to bring the kernel up.
- f. Copies PROLOG to \$USER. If you do not have a PROLOG file created for DMROOT, a simple one can be copied from the release catalog to the \$USER catalog. The PROLOG file is executed every time a job is executed from user name DMROOT. You may answer YES or NO to this prompt.

NOTE

All IM/DM users (including DMROOT) must have the following command in their NOS/VE prolog:

```
:family-name.dmroot.dm$commands
```

where "family-name" is the name of the family that contains the IM/DM binaries. This command provides access to all IM/DM modules. The command is in the .DMROOT.RELC3.PROLOG file.

- g. Copies DM\$COMMANDS to \$USER. This step copies the file DM\$COMMANDS from the release subcatalog to \$USER. You should answer YES to this prompt.
- h. Installs as default release. This step installs the new release as the default release by copying the file DMCMD_COM to the \$USER catalog. You should answer YES to this prompt.
- i. Adds the DM commands. This step will add the DM commands to the command list for your job. No user response is required. Log out and log back in under DMROOT. All IM/DM modules and the ACME database (described in the IM/DM reference manuals) can now be accessed. IM/DM installation is now complete.
- j. Starts up the kernel. This step starts the kernel by running DMSA and issuing the BEGIN command. You may answer YES or NO to this prompt. If you answer YES, this step will bring up the kernel. If you answer NO, the kernel will not be started up in this step. If NO is selected, refer to Initiation of the DM Kernel for information on how to start up the DM kernel.

20.1.2 IM/DM Upgrade

If IM/DM is already running on your current system, you must:

1. Backup your authority database
2. Delete your current IM/DM binaries
3. Proceed to step 2 of IM/DM installation.

20.1.3 Backup Authority Database

The authority database (ADB) contains the information on users registered at your site and databases created at your site. To back up your ADB, use the BACKUP/ADB command in DMSA and back up the ADB to a subcatalog other than DMROOT.RELC3. The suggested method is to back up the ADB to \$USER under DMROOT. After the upgrade is completed, you can use the RESTORE/ADB command in DMSA to restore your copy of the ADB to its proper subcatalog. Refer to the IM/DM Data Administration manual for documentation on the BACKUP/ADB and RESTORE/ADB commands.

20.1.4 Deletion of IM/DM Binaries

To save permanent file space and to assure correct file permissions for the new binaries, delete the current IM/DM binaries.

NOTE

Check that there are no other files stored in the :family-name.DMROOT.RELC3 subcatalog other than the IM/DM binaries, as all files in that subcatalog will be deleted.

Type in the following at the console:

```
task r=4
$system.osf$site_command_library.delcc :family-name.dmroot.relc3
delete_catalog :nve.dmroot.relc3
taskend
```

??where "family-name" is??

20.1.5 Initiation of the DM Kernel

Use the DMSA command to start up the DM kernel. You will be prompted for the user ID and password.

For the user ID, enter:

SAUID

For the password, enter:

SAUPW

You will then be asked if you want to review the user aids for the program. You may enter YES or NO. You will then receive the DMSA prompt. In response, enter the DMSA command:

BEGIN K=1,PW=PW

NOTE

The DMSA BEGIN command requires a password associated with the kernel. The password for all kernels is PW.

You will then receive another DMSA prompt. In response, enter:

exit

Control will then be returned to the operating system. Even though you received control back from the DMSA BEGIN command and left the DMSA module, it may still take several minutes for the kernel to complete its initialization.

After the kernel has completed its initialization, the two batch jobs IM_DM_xx_K01 and IM_DM_xx_S11 (where "xx" represents the release level, for example C2 or C3) should always be seen executing in the system.

20.1.6 IM/DM Installation Site-defined Family Name

At this release, IM/DM supports the NOS/VE feature of allowing a site to specify a site-defined family name. At the time of installation, the installer will supply the family name. The family name will then be placed into all of the appropriate file paths. The installer will supply the family name in two different steps during the installation.

The first place will be during the execution of the UPGRADE_SOFTWARE utility. A new procedure, DEFINE_IM_DM_FAMILY, has been added to the UPGRADE_SOFTWARE utility to allow for the installation of the IM/DM product to a family other than the one registered by the installation table. DEFINE_IM_DM_FAMILY will change the installation table by replacing the current family name of the IM/DM file paths with the one specified by the installer. INSTALL_PRODUCT can then be used to install the IM/DM product under the new family.

The second place will be during the execution of the SCL INSTALL procedure which is run after all of the INSTALL_PRODUCT subcommands have completed. The SCL INSTALL procedure will prompt the installer for the family name

during step 5. The procedure will then change all of the necessary files and file paths by replacing the current family name with the one specified by the installer.

If a site chooses to install IM/DM under a family other than "NVE" at the time of this release, all file paths relating to all existing databases will have to be modified also. The DMSA ASSIGN command can be used to change the file path for the Definition Data Base (DDB) in the Authority Data Base. The DMDBA module can then be used to change all of the file paths in the Structural Data Model (SDM).

20.2 Significant Problems

The following items are significant problems in the area of IM/DM:

1. TERMINATE_JOB Command Can Cause Damaged IM/DM Databases
2. Erroneous Diagnostic When FIXED Area Runs Out of Space
3. TERMINATE_JOB command Damages IM/DM Databases
4. Delay in Output of DM Modules
5. Conversion Error on Record Containing Negative Number
6. DISCARD Result Sets Rolls Over at 65771
7. FIND Command in Certain Cases Result in an Error

20.2.1 TERMINATE_JOB Command Can Cause Damaged IM/DM Databases

There is an intermittent problem with databases becoming damaged if the kernel is terminated via a NOS/VE TERMINATE_JOB command while the database still has a transaction in progress. When this situation occurs, the database is marked as damaged and is unrecoverable. An additional problem is that the kernel will not issue a diagnostic to inform the user that the database is unrecoverable when recovery is being performed on the subsequent kernel initiation. No diagnostic is issued until an attempt to access the database is made. The only known workaround for this problem is to avoid use of the NOS/VE TERMINATE_JOB command to terminate the kernel if any databases are open which have transactions in progress. If your site typically lets the IM/DM kernels run until the system is brought down by the computer operator, it would be advisable to have the operator terminate the two IM/DM kernel jobs last. The database must be restored from a previous backup if this problem occurs.

20.2.2 Erroneous Diagnostic When FIXED Area Runs Out of Space

There is a problem with an application program getting an erroneous diagnostic if the program is adding view occurrences into a FIXED area of an occurrence database and the FIXED area runs out of space. You can determine whether or not you have a FIXED area by checking the ALLOCATION parameter on the appropriate AREA statement in the Structural Data Model. The erroneous diagnostic which is issued is "informational error 1 - End of information encountered." The diagnostic which should be issued is one which tells you that the AREA is full. The workaround for the erroneous diagnostic problem is to use DYNAMIC areas instead of FIXED areas. If you run out of space in a DYNAMIC area, the correct diagnostic will be issued. The workaround for the problem of running out of space in any area (FIXED or DYNAMIC) is to increase the value for the PAGES_IN_AREA parameter on the AREA statement.

PSR DM8A163

20.2.3 TERMINATE_JOB command Damages IM/DM Databases

There is a problem that occurs when an application update program is interrupted either when the kernel is taken down or the program is ended by means of the TERMINATE_JOB command. Subsequent jobs accessing the affected database with read intent experience no problems. However, when a DMFQM job attempts to delete records which have been successfully read with a previous FIND command, an error is returned, "NO SUCH PAGE IN FILE - END OF FILE DETECTED BY A DIRECT READ". When the database is closed and subsequently opened, the user gets a message that one of their database files is damaged.

PSR DM80153.

20.2.4 Delay in Output of DM Modules

DM modules (e.g. DMDBA) write output to a terminal as it is produced, but the output is buffered by NOS/VE and is written to the terminal only when the buffer is full. Hence, an interactive user may experience delays in seeing output appear on the screen. This is not an IM/DM problem. The user can type "escape-E" to force the buffer to be written to the terminal; this has no effect on processing by IM/DM.

PSR DM80211.

20.2.5 Conversion Error on Record Containing Negative Number

There is a problem with adding a record that contains a negative number to a database when using a COBOL form view and an output format with the TSO (Trailing Sign Overpunch) option. A data conversion error occurs and DMSTAT 780 is returned.

PSR DM80227

20.2.6 DISCARD Result Sets Rolls Over at 65771

There is a problem with the DISCARD command to release active result sets. When the set number exceeds 65771, the next set number rolls over to 1 and does not point to a retrieved set. An attempt to reference the next set returns an error that set 1 does not exist.

PSR DM8A208

20.2.7 FIND Command in Certain Cases Result in an Error

There is a problem with using the FIND command with a composite view and sorting the generated result set. In certain cases, error 815, "RECORD NOT FOUND" is returned.

PSR DM80217

20.3 Notes and Cautions

The following items are notes and cautions in the area of IM/DM:

1. File Descriptors May Be Represented by Logical File Names
2. Use PADDING Statement to Create Indexes
3. Use Header Line Information for Submitting PSRs
4. Information Required with PSRs
5. New Version of the FIXC2 Utility for IM/DM

20.3.1 File Descriptors May Be Represented by Logical File Names

Internal IM/DM file descriptors may sometimes be represented by logical file names. This may cause some confusion to the user. For example, the NOS/VE DISPLAY_COMMAND_INFORMATION command, when used on an IM/DM module, will show that some file descriptors in the display may not be standard NOS/VE file

descriptors. The value is a logical file name and will be meaningless to the user until it is translated into a standard NOS/VE file descriptor. To translate any of these logical file names and determine the actual file path for a given file, an internal IM/DM SCL procedure called `DISPLAY_LOGICAL_NAME` can be used. This SCL procedure is in one of the standard IM/DM libraries (`:NVE.DMROOT.RELC3.SLANG.PROC_LIB`) and is available to all IM/DM users. The `DISPLAY_LOGICAL_NAME` command accepts a logical file name as input and will display the value of that logical name as output. See the IM/DM Data Administration Usage Manual (60489014) for more information on logical file names.

20.3.2 Use PADDING Statement to Create Indexes

When using the DMR utility to create indexes, the use of the "PADDING" parameter on the INDEX statement in the Structural Data Model is strongly recommended. The PADDING parameter is used to specify how much space should be reserved for future updates to the index. It should especially be used if there are many index terms or if the database is expected to be updated frequently. The actual value to be used for padding is dependent on the expected amount of updating. For use with DMR, a non-zero padding factor should always be specified. The more updating expected after the DMR run, the larger the padding factor should be. A value of 5 to 10 is recommended. Please note that a padding value will lead to a larger file. Consequently, if DYNAMIC areas are being used, a large value should be specified for the `PAGES_IN_AREA` parameter. Specifying a large padding value will help avoid performance degradations during DMR index creations.

20.3.3 Use Header Line Information for Submitting PSRs

Each IM/DM module has a header line which is displayed when the module is invoked. This header line contains the release level information at the end of the line. This release (and all subsequent releases) will include both the product level and the operating system level as part of the release level information which appears in the header line. For example, this release shows a release level of "C3C121". This information should be used for the product level field in submitting PSRs and should also be included in the inquiry text.

20.3.4 Information Required with PSRs

This section describes what information is necessary to be submitted with a PSR. There are four situations covered as follows: general problems, kernel crashes, sort kernel crashes, and kernel hangs.

1. For any general problem the following is a list of support materials we would like to have:

- a. A description of the circumstances in which the problem occurs.
- b. The names of the database, user data model and files involved. A listing of the ADM, SDM, and UDM is also required.
- c. A tape dump of the database files and any program files involved (if practical) including a listing of which files are on the tape and information on the tape density and format.
- d. If the problem has occurred in a user application program, a listing of the program source, compilation, and execution, including IM/DM and host status codes and any program job logs. If other steps were required before the program was run, a description of the steps involved would be helpful. If the problem has occurred in interactive mode, a listing of the interactive session, beginning with the IM/DM sign-on.
- e. An indication of whether or not the problem is reproducible.
- f. If the problem indicates that there might be information in the kernel job log or that the nature of the problem is related to a permanent file, a listing of the kernel job log should be provided. This can be found in the following file:

:NVE.DMROOT.RELC3.DM.RUNDMK.JOB_LOG_01

2. If the kernel crashes, there are four files in the :NVE.DMROOT.RELC3.DM.RUNDMK subcatalog which will be important in determining the cause of the crash. Please check the creation date and time on these files using the DISPLAY_CATALOG_ENTRY command to be sure that you are copying the correct files. If any of these files were not created at the time of the crash please indicate this in the PSR. These files must be copied either to new file names in the same subcatalog or to files in another subcatalog before the kernel is brought back up. This is necessary because the kernel will reuse the same files over again the next time it is brought up. The file names in the RUNDMK subcatalog which must be copied are as follows:

JOB_LOG_01
DMK_LOAD_MAP_01
DMK_CALLS
DUMPFIL1_DMX

It may be helpful to include the date and time of the crash as part of the new file name. An example of this follows:

```
SETWC $USER.RELC3.DM.RUNDMK
COFF JOB_LOG_01      $USER.JOB_LOG_01_041586_0855
COFF DMK_LOAD_MAP_01 $USER.DMK_LOAD_MAP_01_041586_0855
COFF DMK_CALLS       $USER.DMK_CALLS_041586_0855
COFF DUMPFIL1_DMX    $USER.DUMPFIL1_DMX_041586_0855
```

In the event of a kernel crash, these four files should then be copied to tape and submitted with a PSR. In addition, as many as possible of items (a) through (e) in section 1 above should be supplied. Again, a listing of which files are on the tape and information on the tape density and format should be included.

3. If the sort kernel crashes, the same information which is required for a kernel crash is required. However the files which should be sent are different. The file names which are required for a sort kernel crash are also in the RUNDMK subcatalog, but the names are as follows:

JOB_LOG_01
JOB_LOG_11
DMSMK_LOAD_MAP_11
DMSMK_CALLS

In the event of a sort kernel crash, these four files should be copied to tape and submitted with a PSR. In addition, as many as possible of items (a) through (e) in section 1 above should be supplied. Again, a listing of which files are on the tape and information on the tape density and format should be included.

4. If the kernel hangs, the same information that is required for a crash is required. However, the user should force a dump to be produced by using the DMSA DUMP command. If the DMSA DUMP command will not work, the user should use the DMSA DESTROY command. The DESTROY command will take a dump, and then force the kernel to come down, regardless of whether or not there are any users signed on. It should only be used to bring the kernel down as a last resort. Please refer to the IM/DM Data Administration Manual (60489014) for documentation on the DMSA DUMP and DESTROY commands.

20.3.5 New Version of the FIXC2 Utility for IM/DM

If you are upgrading from IM/DM C2A to IM/DM C3A under NOS/VE 1.2.1 L670 (i.e. you did not install the asynchronous L659 IM/DM C3A against NOS/VE 1.1.4 L649) and you wish to use the FIXC2 utility to convert your databases, you will need to request a copy of FIXC2. The FIXC2 utility was documented in the L659 SRB/FNB document. It was a tool which could be used to automatically convert C2A databases to C3A databases, as there was an incompatibility between these two releases of IM/DM. To obtain a NOS/VE 1.2.1 L670 version of the FIXC2 utility, please contact Central Software Support.

Chapter 21

IM/Quick

21.1 Significant Problems

The following items are significant problems in the area of IM/Quick:

1. Unpredictable Results When Output Format Is Incorrect

21.1.1 Unpredictable Results When Output Format Is Incorrect

If an IM/DM data model specifies that the output format of an element contains more digits to the right of the decimal point than the total output format will accomodate, IM/Quick's results are unpredictable. As an example, for an output picture of >2.2>, in one case IM/Quick issued the following erroneous diagnostic:

This input description must be to match the description
in the definition data base.

In another case, IM/Quick did not diagnose an error and generated 99 as the output picture.

The work-around is to correct the IM/DM data model so that the total output format contains more character positions than the number of digits to the right of the decimal point.

PSR QK8A018

Chapter 22
LISP

22.1 Notes and Cautions

The following items are notes and cautions in the area of LISP:

1. TERMINATION_ERROR_LEVEL Set to FATAL

22.1.1 TERMINATION_ERROR_LEVEL Set to FATAL

The TERMINATION_ERROR_LEVEL in the program description for LISP/VE is currently set to fatal and must not be changed.

Chapter 23 Magnetic Tape Usage

23.1 Notes and Cautions

The following items are notes and cautions in the area of Magnetic Tape Usage:

1. Unsupported Program Interfaces
2. adding Character for EBCDIC Labelled Tape Files
3. Maximum WORKING_STORAGE_LENGTH for Tape Files
4. Labelled Tape File Block and Record Type Restrictions
5. Maximum Block Length Restrictions for Tape Files

23.1.1 Unsupported Program Interfaces

The AMP\$SKIP program interfaces is not completely supported for magnetic tape files for NOS/VE 1.2.1 L670, specifically for the following cases:

1. `skip_unit = amc$skip_record` and `skip_direction = amc$forward` is not implemented for BT=SS with RT=V. It is not allowed for BT=SS with RT=U.
2. `skip_unit = amc$skip_record` and `skip_direction = amc$backward` is not implemented for BT=US with RT=F and BT=SS with RT=V. It is not allowed for BT=SS with RT=U.
3. `skip_unit = amc$skip_partition` and `skip_direction = amc$forward` or `skip_direction = amc$backward` is not implemented for BT=SS with RT=V. It is not allowed for other BT and RT combinations.
4. `skip_unit = amc$skip_tapemark` and `skip_direction = amc$forward` or `skip_direction = amc$backward` is supported for all BT and RT combinations for unlabelled and non-standard labelled tape files. It is not allowed labelled tape files.

23.1.2 adding Character for EBCDIC Labelled Tape Files

When writing ANSI Fixed records (BT=US with RT=F) to a labelled tape file with a `character_set` of EBCDIC and `character_conversion` value of TRUE, the record padding characters are not converted to EBCDIC tape and therefore are read back as semicolons (;). To avoid the problem, specify a padding character of '@' (commercial at sign) on the `CHANGE_TAPE_LABEL_ATTRIBUTES` command for the tape file. This will cause the padding characters to be

written to the tape as blanks (spaces). To avoid a similar problem with block padding, specify the `maximum_record_length` as an exact multiple of the `maximum_block_length` on the `CHANGE_TAPE_LABEL_ATTRIBUTES` command.

23.1.3 Maximum `WORKING_STORAGE_LENGTH` for Tape Files

The maximum `working_storage_length` that is supported for magnetic tape files with a `block_type` of `system_specified` and a `record_type` of `variable` is 123,840 bytes. The maximum value for `working_storage_length` is unlimited for the other tape file `block_type` and `record_type` combinations.

23.1.4 Labelled Tape File Block and Record Type Restrictions

Labelled tape files are not supported for tape files with a `block_type` of `system_specified` and a `record_type` of `variable`. Labelled tape support is provided for the other tape file `block_type` and `record_type` combinations.

23.1.5 Maximum Block Length Restrictions for Tape Files

The maximum block length allowed for all block and record combinations for systems configured with a single PP tape driver is 4128 bytes. For systems configured with a dual PP tape driver, the maximum block length depends on the `block_type`, `record_type`, page size and real memory available to the job.

BT=US RT=U	2M bytes with a 4K page size	8M bytes with a 8K page size, 32M bytes with a 16K page size
BT=US RT=F	2M bytes with a 4K page size, 8M bytes with a 8K page size, 32M bytes with a 16K page size	
BT=SS RT=U	4128 bytes	
BT=SS RT=V	4128 bytes	

Chapter 24

MAIL/VE

MAIL/VE is an electronic mail utility used to read and receive letters between mailboxes.

24.1 Installation

NOTE

MAIL/VE should be installed only once. On subsequent upgrades of NOS/VE, you must dump the entire subcatalog `$SYSTEM.MAILVE.MAIL` before you terminate the old system, then reload this subcatalog on the new system. This subcatalog contains all the files necessary to resume operations on the new system.

To install MAIL/VE:

1. From the NOS/VE system console, enter:

```
set_command_list add=$system.mailve.maintenance.command_library
```

This adds the MAIL/VE system administrator command library to your command list.

2. From the NOS/VE system console, enter:

```
install_mail family=family-name
```

where "family-name" is a NOS/VE family.

The `INSTALL_MAIL` command builds the MAIL/VE database files and registers the first MAIL/VE users: `$SYSTEM` and `MAILVE`.

3. Create the user name `MAILVE` on NOS/VE. Also, create the user name `MAILVE` on the NOS or NOS/BE system, if your site operates in dual-state.
4. Log in under user name `MAILVE` on NOS/VE and execute the following commands:

```
set_working_catalog $user
mail
create_catalog mail
create_distribution_list list_name=dummy_list address_list=mailve
delete_distribution_list list_name=dummy_list
quit
```

Execution of these commands gives the MAIL/VE user access to all of the files within the mail database.

24.2 MAIL/VE Maintenance

The MAINTAIN_MAIL utility is used for MAIL/VE maintenance. Its subcommands register and delete users, add and delete families, and display mail statistics. To execute the utility, you must be under user name MAILVE or at the NOS/VE system console. MAINTAIN_MAIL and all its subcommands require the password VUNY. To execute the MAINTAIN_MAIL utility, enter:

```
set_command_list add=$system.mailve.maintenance.command_library  
maintain_mail pw=vuny
```

While in the utility, you can access the online manual for MAINTAIN_MAIL by entering:

```
help
```

You can also access the online manual by entering the command:

```
explain m=$system.mailve.maintenance.maim_ve
```

Refer to this online manual for more information on MAINTAIN_MAIL and the MAIL/VE maintenance procedures.

24.2.1 MAIL/VE Catalogs and Files

After the MAIL/VE installation, the following catalogs and files are created under catalog \$SYSTEM.MAILVE.

BOUND_PRODUCT	File containing the bound MAIL/VE binary.
MAIL	Catalog where the current database files are stored.
BACKUP	Catalog where backup copies of the database files are stored in the file BACPF_FILE.i]
MAINTENANCE	Catalog that has the following files in it.
	COMMAND_LIBRARY MAIL/VE system administrator's command library.
MAIM_VE	MAIL/VE system administrator's online manual.

24.3 Significant Problems

The following items are significant problems in the area of MAIL/VE:

1. Changing Family Names with MAIL/VE
2. Check Mail Fails on WYSE-60
3. The Cursor Is Left in an Unexpected Position After a DELETE

24.3.1 Changing Family Names with MAIL/VE

Moving mailboxes from one family to another must be done by manually redefining each mailbox on the new family. Old mail on the old family may still be accessed as long as the old mailboxes are left with the old family name. Reading mail from the old family is done by changing the mailbox name at the top of the CHECK_MAIL screen from MAILBOX_NAME to :old-family.MAILBOX_NAME, where "old-family" is the old family name.

PSR NV01778

24.3.2 Check Mail Fails on WYSE-60

CHECK_MAIL does not work in screen mode on a WYSE 60 terminal because the cursor cannot be positioned on the desired letter. This terminal model prevents the cursor from entering protected fields.

PSR MVE0047

24.3.3 The Cursor Is Left in an Unexpected Position After a DELETE

After a letter is deleted from the CHECK_MAIL screen, the cursor is returned to the first letter in the list. The user must take care that the next delete entered will delete the letter on which the cursor rests, whether or not the letter has been read.

PSR MVEA053

24.4 Notes and Cautions

The following items are notes and cautions in MAIL/VE:

1. BACKUP_MAIL and RESTORE_MAIL Procedures

24.4.1 BACKUP_MAIL and RESTORE_MAIL Procedures

Procedures BACKUP_MAIL and RESTORE_MAIL require exclusive access to the data base files and therefore cannot be run from MAINTAIN_MAIL which also requires exclusive access.

A change in the NOS/VE file sharing at Level 664 caused the BACKUP_MAIL and RESTORE_MAIL commands to fail. As the current procedures are somewhat nonstandard, the contents were replaced by code which uses the NOS/VE BACKUP and RESTORE utilities. The MAIL/VE backup file is now placed in \$SYSTEM.MAILVE.BACKUP.BACPF_FILE, and this file contains the contents of each MAIL/VE database file. Formerly each database file was copied to a separate file on the backup catalog.

Parameters for the BACKUP_MAIL procedure are unchanged. However, the CREATE_NEW_FILE_CYCLES parameter has been removed from the RESTORE_MAIL procedure, as ??RESTORE_MAIL?? always writes the new backup file to a new cycle.

Chapter 25
MALET/VE

25.1 Significant Problems

The following items are significant problems in MALET/VE:

1. Buffer Controller Memory Test MY8 Fails
2. MALET/VE Recovery Fails
3. Random Busy Errors Can Occur Running DTI

25.1.1 Buffer Controller Memory Test MY8 Fails

Module 02 of the buffer controller memory test (MY8) fails when run in the sequence of modules. It does, however, run by itself. To avoid this problem, module MY802 should be bypassed when run in a sequence. For example: RUN MY8 (00,01). And explicitly run: RUN MY802.

PSR MAVAO09

25.1.2 MALET/VE Recovery Fails

The MALET/VE recovery of a hung PP, MLD/VE was residing in, is not released yet. Any reference to recovery should be ignored until this feature is available in the next release.

25.1.3 Random Busy Errors Can Occur Running DTI

When running DTI on dual channel 895s, random "Busy" errors (General Status = 0002) can occur if this MALET/VE test is running through one channel while NOS/VE is accessing the 895s through the other channel.

PSR MAVAO15

25.2 Notes and Cautions

The following items are notes and cautions in MALET/VE:

1. MALET/VE Requires a Minimum 4K Page Size

25.2.1 MALET/VE Requires a Minimum 4K Page Size

At NOS/VE 1.2.1 L670, a page size of 4K or greater is required if MALET/VE is to be used at your site.

Chapter 26
NAM/VE

26.1 Manual Errata

The following items are manual errata for the NOS/VE Network Interface Usage Manual (60463916):

1. SYSTEM_PRIVILEGE and CAPABILITY Parameters Have No Effect

26.1.1 SYSTEM_PRIVILEGE and CAPABILITY Parameters Have No Effect

The reference to the SYSTEM_PRIVILEGE and CAPABILITY parameters for the CHANGE_CLIENT_VALIDATION and CHANGE_SERVER_VALIDATION subcommands of the DEFINE_CLIENT and DEFINE_SERVER commands should be ignored. These parameters have no effect at NOS/VE 1.2.1 L670.

Chapter 27 Operations

27.1 Notes and Cautions

The following items describe operational notes and cautions:

1. VED IJD Operator Display

27.1.1 VED IJD Operator Display

Appendix E of the SRB describes the VED IJD operator display, which displays detailed status information for all initiated jobs.

27.2 Manual Errata

The following are manual errata affecting operations:

1. Two New Operator Displays for VEDISPLAY
2. VEDISPLAY DS
3. VEDISPLAY PP
4. Changing Family Names with MAIL/VE
5. Tape Commands
6. CHANGE_DATE (CHAD) Command
7. CHANGE_TIME (CHAT) Command

27.2.1 Two New Operator Displays for VEDISPLAY

The DISPLAY_OPTION parameter for the VEDISPLAY command has the additional values:

PP Displays information about the PPs actively configured to NOS/VE.
DS Displays information about the storage devices actively configured to NOS/VE.

A description of these two displays follows.

27.2.2 VEDISPLAY DS

The VED DS command displays information about the storage devices actively configured to NOS/VE. Each entry of the display has the following format:

name product state unit channel(s) vsn qc mac lun

name	The first 14 characters of the storage device's name in the NOS/VE physical configuration. If the name exceeds 14 characters, the name is suffixed by '..'.
product	The first 6 non-blank characters of the storage device's element identification exclusive of any leading '\$'.
state	The state of the storage device in the physical configuration, e.g. ON, OFF or DOWN.
unit	The octal unit number of the storage device.
channel(s)	The name of up to two channels through which the storage device may be accessed by NOS/VE. A concurrent (CIO) channel in an I4 is identified by CCHn and all other channels are identified by CHn, where n is the decimal representation of the channel number.
vsn	The recorded_vsn which identifies the media mounted on a mass storage device and the external_vsn which identifies the media mounted on a tape storage device.
qc	The number of physical i/o requests currently on the queue for the storage device.
mac	This field identifies whether there is any maintenance access occurring which is referencing this storage device. An integer value indicates the number of jobs which have requested CONCURRENT maintenance access. The value 'D' indicates one job has requested DEDICATED maintenance access.
lun	This is the NOS/VE internal identification of the storage device for an instance of deadstart. Each storage device has a unique number, i.e. a Logical Unit Number.

27.2.3 VEDISPLAY PP

The VED PP command displays information about the PPs actively configured to NOS/VE. Each entry of the display has the following format:

iou pp state ssn iou-program channel

iou	The name of the IOU to which the PP belongs. This is currently always the value 'IOU0'.
pp	The value 'PPn' is displayed for all PPs except the concurrent (CIO) PPs in an I4 which are identified by 'CPPn', where 'n' is an octal value.
state	The state of the PP (ON, OFF, or DOWN). This is currently always the value 'ON'.
ssn	The system-supplied-name of the job to which the PP has been reserved. If the PP is reserved to a NOS/VE system task, the value '\$SYSTEM' is displayed.
iou-program	The name of the program executing in the PP.
channel	The octal channel number to which the PP has been assigned. A concurrent (CIO) channel in an I4 is identified by a 'C' next to the channel number.

27.2.4 Changing Family Names with MAIL/VE

The Restoring Permanent Files to a Different Family section of chapter 10 of the NOS/VE Operations Manual (60463914) lists several items to consider if you are changing your NOS/VE family name. Add the following consideration to this list.

Moving mailboxes from one family to another must be done by manually redefining each mailbox on the new family. Old mail on the old family may still be accessed as long as the old mailboxes are left with the old family name. Reading mail from the old family is done by changing the mailbox name at the top of the CHECK_MAIL screen from MAILBOX_NAME to :old-family.MAILBOX_NAME where old-family is the old family name.

PSR NV01778

27.2.5 Tape Commands

1. If the operator wishes to stop a tape request (for example, when the requested tape is not available), the operator can enter

ASSIGN_DEVICE job-name STOP

where "job-name" is the name of the job requesting the tape. This will cause the tape request to disappear and the task requesting the tape to receive the message:

--ERROR RM 881-- OPERATOR TERMINATED TAPE ASSIGNMENT

Note that an interactive user may also terminate a tape request by entering a user break.

2. If the operator mounts the tape on one unit, and does an ASSIGN_DEVICE to another tape unit, the operator will be requested to ready the other tape unit. Rather than move the tape to the other unit, the operator may enter:

ASSIGN_DEVICE job-name REASSIGN

where "job-name" is the name of the job requesting the tape. This will cause the tape request to reappear, then the operator can specify the correct unit. When a tape has been assigned to a unit that cannot be made "READY", this command can be used to reassign the tape to another unit.

27.2.6 CHANGE_DATE (CHAD) Command

Purpose: Changes the current date.

Format: month, m : integer 1 .. 12 = \$required
day, d : integer 1 .. 31 = \$required
year, y : integer 1900 .. 2155 = \$required
status

Remarks: This command is only valid for standalone systems. For dual-state systems, changing the date on the host system (NOS or NOS/BE) causes the date to be changed on the NOS/VE system.

Example: The following changes the system date to September 12, 1986:

change_date year=1986 month=9 day=12

27.2.7 CHANGE_TIME (CHAT) Command

Purpose: Changes the current time.

Format: hour, h : integer 0 .. 23 = \$required
minute, m : integer 0 .. 59 = \$required
second, s : integer 0 .. 59 = 0
status

Remarks: This command is only valid for standalone systems. For dual-state systems, changing the time on the host system (NOS or NOS/BE) causes the time to be changed on the NOS/VE system.

Example: The following changes the system time to 11:30 AM:

```
change_time hour=11 minute=30
```

Chapter 28
PASCAL

28.1 Significant Problems

The following items are significant problems in the area of PASCAL:

1. Nested Variables May Not Be Valid in Outer Procedure/Function

28.1.1 Nested Variables May Not Be Valid in Outer Procedure/Function

When variables which are local to a procedure/function, but global to a nested procedure/function are modified by the nested routine, there exists a possibility of using a stale reference to such a variable in the enclosing procedure/function.

PSR PA80033

Chapter 29

Permanent File Transfer Facility

29.1 Significant Problems

The following items are significant problems in the area of Permanent File Transfer Facility:

1. PTF Requires Family Names of at Least Three Characters
2. Permanent File Transfer Facility Concurrency Problem
3. Error Notification for User Prolog Problems

29.1.1 PTF Requires Family Names of at Least Three Characters

The Permanent File Transfer Facility (PTF) will not work correctly if the family name is less than three characters.

29.1.2 Permanent File Transfer Facility Concurrency Problem

In PTF level 670 users may have some difficulties using PTF to access remote files. The problem is really a side effect attaching files to a job, and not detaching them. To understand this problem better, a little background is in order.

The NOS/VE PTF implementation involves a client task (the user) and a server job. When a user executes a `MANAGE_REMOTE_FILES` command (or an implicit remote file access such as `COPY_FILE`) a server job is started on the remote host on the user's behalf. The server job executes SCL commands for the user. When the SCL commands for the user have been completed, or an error has terminated SCL processing, the server job completes protocol with the client task and begins termination. Job termination will return all files attached by the user via SCL, however job termination is not instantaneous. It is possible the user on the client machine may execute another `MANRF` command and have the server start on the remote host before the previous server is terminated. If the second server attempts to access files still attached by the first server, an error will likely occur. For example:

```
/MANRF L=REMOTE_HOST F=$NULL
mrf? LOGIN USER PASSWORD
mrf? ATTACH_FILE X
mrf? **
/MANRF L=REMOTE_HOST F=$NULL
mrf? LOGIN USER PASSWORD
```



```
mrf? ATTACH_FILE X          "file may still be attached"
mrf? **
```

It is important to note that job termination time is variable. It is dependent primarily on three things: user epilogs, system epilogs, and the amount of activity on the mainframe. User epilogs which copy the amount entire session job log to a permanent file will delay termination substantially.

A more subtle side of this problem is implicit file attaches by SCL. Many SCL commands cause a file to be attached to the job but not detached. For example:

```
/MANRF L=REMOTE_HOST F=$LOCAL.X
mrf? LOGIN USER PASSWORD
mrf? RECEIVE_FILE Y
/MANRF L=REMOTE_HOST F=$LOCAL.Z
mrf? LOGIN USER PASSWORD
mrf? SEND_FILE Y
mrf? **
```

In this case the file Y on the remote host is implicitly attached to the job. If the first server job on the remote host is not terminated before the second server job attempts to use the file, an error will occur. It is expected that implicit attach/detach will be provided in a future NOS/VE release (PSR NVOE322). When this feature is available, the file Y will be detached in the first job when the RECEIVE_FILE command is complete.

29.1.3 Error Notification for User Prolog Problems

If a user prolog has an error, PTF server will be unable to execute SCL commands on behalf of the user. Additionally, no information on the error can be returned to the initiating user. However, the job log from the failed PTF server job will be printed on the remote system.

29.2 Manual Errata

The following are manual errata affecting Permanent File Transfer Facility:

1. Error NF4 on MANRF or PTF File Access
2. Able to Receive Files But Not Send or Vice Versa

29.2.1 Error NF4 on MANRF or PTF File Access

If the error

```
--ERROR NF 4-- Service is not established with location family-name.
```

where "family-name" is a family you referenced on an implicit PTF file access command or through `MANAGE_REMOTE_FILE`, then check that the family was specified correctly and that the remote host for that family is running.

If both of these are true, then you should insure that the PTF system task is running on the remote host. You do this by entering the following command from the remote host's operator console.

```
display_active_tasks
```

and determine if the system task `NFP$FILE_TRANSFER_BOOT` is listed. If this entry is not listed, you can activate the task by entering:

```
activate_ptf
```

If the system task is listed, enter:

```
manage_network_application  
display_server_attributes osa$file_transfer_server
```

and check the display output in the `TITLES` section. The titles available for connect are of the form `PTFSS$x` where `x` is one or more characters. The server is connected to by specifying the `x` part of the title. If the title(s) match what was expected, it is likely that the physical connection between the two hosts is down.

29.2.2 Able to Receive Files But Not Send or Vice Versa

If, from Machine A, you can send files to Machine B, but cannot receive files from Machine B, then the server task probably is not executing on Machine A. To verify this, enter:

```
display_active_tasks
```

from Machine A's operator console, and determine if the system task `NFP$FILE_TRANSFER_BOOT` is listed. If this entry is not listed, you can activate the task by entering:

```
activate_ptf
```

29.3 Manual Errata

The following are manual errata affecting Permanent File Transfer Facility:

1. Pause Break on PTF Ignored

29.3.1 Pause Break on PTF Ignored

A pause break entered during a PTF transfer is ignored, except when entered through an INTERCOM connection, in which case the transfer is terminated.

Chapter 30 Programming Environment

30.1 Notes and Cautions

The following items are notes and cautions in the area of Programming Environment:

1. Creation of C Programs in the Programming Environment

30.1.1 Creation of C Programs in the Programming Environment

The two characters "_C" are appended to the names of all C programs created in the Programming Environment. This occurs when one is using the CREATE and IMPORT function of the Programming Environment to create a C program and the name of the program supplied by the user does not end with the characters "_C". This means that the user may supply the first 29 characters of the name of a C program, or 31 if it ends with the characters "_C", instead of the 31 characters as specified in the online manual.

Chapter 31
PROLOG

31.1 Notes and Cautions

The following items are notes and cautions in the area of PROLOG:

1. Many Predicates Are Now Defined with Arity 2

31.1.1 Many Predicates Are Now Defined with Arity 2

With NOS/VE 1.2.1 L670 many of the Prolog I/O predicates, which were previously defined only with arity 1, are now defined both with arity 1 and with arity 2. Programs which use these I/O predicates with arity 1 need not be changed. In 1.1.4, defining one of them with arity 2 would not have been an error; with NOS/VE 1.2.1 L670 it will result in

--ERROR-- Attempt to redefine a system predicate.

Chapter 32 Screen Design Facility

32.1 Notes and Cautions

The following items are notes and cautions in the area of Screen Design Facility:

1. Intensity Attribute
2. Integer and Real Variables
3. User Specified Names Must Be SCL Names

32.1.1 Intensity Attribute

Specification of the high or low intensity attribute will result in the intensity being set to the alternate of its current setting.

PSR NVOJ886

32.1.2 Integer and Real Variables

Initial values given for integer and real variables are not validated. Further, display of illegitimate values is spurious when presented on the user object screen, as compared to the SDF design screen.

PSR NVO_____

32.1.3 User Specified Names Must Be SCL Names

All user specified names are treated as system names until such time as a screen definition is saved. Names must comply with system naming rules. When a screen is saved with COBOL language specified, underscore characters appearing in the names are converted to hyphens in the screen source record definition. Special characters should not be used in names intended for COBOL use. PSR SD8A002

Chapter 33 Sort/Merge

33.1 Significant Problems

The following items are significant problems in the area of Sort/Merge:

1. MINIMUM_RECORD_LENGTH File Attribute

33.1.1 MINIMUM_RECORD_LENGTH File Attribute

File attribute MINIMUM_RECORD_LENGTH is not set properly for RECORD_TYPE=F files. MINIMUM_RECORD_LENGTH defaults to 0. This affects Sort/Merge when the sort key is omitted and the default key is used. The user should explicitly set MINIMUM_RECORD_LENGTH to the same value as MAXIMUM_RECORD_LENGTH.

PSR NV0D398

33.2 Notes and Cautions

The following items are notes and cautions in the area of Sort/Merge:

1. SMV\$MEMORY_USAGE_LIMIT
2. PAGE_AGING_INTERVAL

33.2.1 SMV\$MEMORY_USAGE_LIMIT

The amount of memory Sort/Merge uses can be specified by the SCL variable SMV\$MEMORY_USAGE_LIMIT. This variable should have JOB scope, and contain an integer value ranging from 64 to 16383. Sort/Merge will use a value which has been rounded up to a multiple of the page size. This value specifies the number of 1024 byte blocks Sort/Merge uses in performing in-core sorts. The default value is 256 pages, whatever the page size might be. For this release, no error is issued for illegal values; they are silently ignored and a value of 64 or 16383 is used. Up to 16 megabytes can therefore be used to sort files, vastly improving the performance of very large sorts. Note: a large value will not hurt performance of small sorts. Sort/Merge will always use only as much as necessary to sort small files.

SMV\$MEMORY_USAGE_LIMIT also affects the merge order used for large sorts. (Note: this variable has absolutely no effect on user-initiated MERGES.) First, by permitting more data to be sorted in memory at one time, fewer sort "strings" will be produced, possibly resulting in fewer merge passes. Also, the maximum merge order goes up possibly resulting in fewer merge passes.

The optimum value for SMV\$MEMORY_USAGE_LIMIT is sensitive to the randomness of the data being sorted. For data that are already mostly in order, small values are preferable to large values. Also, large values can cause very high page faulting and monitor overhead unless the page aging interval is adjusted upwards.

A good way to use this feature is to set this variable in the system prolog. The value can depend on the job mode (smaller values for interactive, larger for batch), and on how much memory the computer has. Individual user prologs or job decks can override this value with whatever they consider appropriate for the sort at hand.

33.2.2 PAGE_AGING_INTERVAL

The job attribute PAGE_AGING_INTERVAL can have a profound impact on Sort/Merge performance. This attribute controls how fast pages are aged from the working set. When SMV\$MEMORY_USAGE_LIMIT is used to increase the amount of memory used and this attribute is not altered, pages are removed too soon and thrashing results. This can be corrected by increasing the aging interval. The optimum value depends on the model. Slow models need a higher value than faster models. For the default of 256 pages, the optimum page aging interval for the CYBER 180/830 is about 500000 microseconds, and for the CYBER 180/860 it is 100000 microseconds. The PAGE_AGING_INTERVAL attribute is altered using the CHANGE_JOB_ATTRIBUTES command, and displayed using the DISPLAY_JOB_ATTRIBUTES command. The PAGE_AGING_INTERVAL parameter of the CHANGE_JOB_ATTRIBUTES command is given in microseconds.

Chapter 34 System Command Language (SCL)

34.1 Significant Problems

The following items are significant problems in the area of SCL:

1. FORMAT_SCL_PROC Occasionally Makes Lines Too Long
2. FORMAT_SCL_PROC Deletes Some Comments
3. Implicit and Explicit Uses of A File Shouldn't Be Mixed
4. \$PROGRAM Function
5. Constants with Unexpected Values
6. Some Local Files Cannot Be Displayed with DISPLAY_CATALOG

34.1.1 FORMAT_SCL_PROC Occasionally Makes Lines Too Long

If there are long parameter or parameter value specifications within the procedure header, the formatter may add spaces and produce a line that is longer than 256 characters. A long procedure declaration should be rechecked after formatting, and leading spaces should be eliminated as necessary.

PSR NVOJ002, NVOI732

34.1.2 FORMAT_SCL_PROC Deletes Some Comments

Any comments contained in a procedure declaration are deleted when formatting takes place.

PSR NVOJ003

34.1.3 Implicit and Explicit Uses of A File Shouldn't Be Mixed

A command that depends on the "path name" for a file is affected if the file has been explicitly attached (ATTACH_FILE, CREATE_FILE commands).

The "path name" that such a command will encounter is the "local path name" rather than the path specified when the command was called. For example:

```
ATTACH_FILE $user.x LFN=y  
REPLACE_FILE $user.x
```

will result in a file called Y being referenced in NOS or NOS/BE rather than X. Users should not mix implicit and explicit use of a file.

PSR NVOD753

34.1.4 \$PROGRAM Function

The \$PROGRAM function does not return the open position designator for the LOAD_MAP file.

PSR NVOD483

34.1.5 Constants with Unexpected Values

The constants \$MIN_INTEGER and \$MAX_INTEGER have values of $-(2^{48}-1)$ and $2^{48}-1$, respectively, rather than $-(2^{63}-1)$ and $2^{63}-1$.

PSR NVOD609

34.1.6 Some Local Files Cannot Be Displayed with DISPLAY_CATALOG

Local files created by a program interface cannot be displayed with DISPLAY_CATALOG. For example, the default files created in FORTRAN do not appear when subsequently entering DISPLAY_CATALOG \$LOCAL. However, executing a command such as REWIND_FILE on the unseen files will cause them to appear with subsequent DISPLAY_CATALOG commands.

RSE NVEA052, NVEA055, NVEA056

34.2 Notes and Cautions

The following items are notes and cautions in the area of SCL:

1. File Routing Requirements
2. Caution to Be Used with Cycle References
3. REDO - Edit Previously Entered Commands
4. Interactive Prompting (SET_COMMAND_MODE)

34.2.1 File Routing Requirements

If a file has `DESTINATION_USAGE=DUAL_STATE` and `DUAL_STATE_ROUTE_PARAMETERS` are supplied, it is important that the user has valid link attributes before attempting `PRINT_FILE`, `CHANGE_JOB_ATTRIBUTES`, or `CHANGE_OUTPUT_ATTRIBUTES`. The link attributes are used as the family name, user name, and password values for the job that routes the file on the dual-state partner. If the link attributes are invalid, the file will be discarded when it is transferred to the dual-state partner. (`SET_LINK_ATTRIBUTES` is documented in the SCL System Interface Manual (60464014).)

34.2.2 Caution to Be Used with Cycle References

Problems may arise if cycle references for a file are not consistent. Consistency in this context means that the cycle references are all generic (e.g., `$high`, `$next`, or omitted entirely), or are all numeric. Inconsistency may result in seemingly inexplicable "cycle busy" or "file still opened" status messages.

Within an SCL procedure, if a file is referenced via an expression containing calls to the `$fname` and `$string` functions, try to arrange to establish a variable containing the string representation of the file near the beginning of the procedure; then use that variable throughout the procedure to reference the file. Not doing this can cause the value returned by the `$string` function to change (with respect to the cycle reference component of a file reference) as the file is accessed, detached, etc.

34.2.3 REDO - Edit Previously Entered Commands

The NOS/VE REDO facility allows you to modify and execute a previously entered command without retyping the entire command. Usage information on this experimental program is in Appendix D.

34.2.4 Interactive Prompting (`SET_COMMAND_MODE`)

For a command entered from an interactive terminal, SCL issues prompting information to the terminal depending upon user actions and the command mode in which the user is operating. The `SET_COMMAND_MODE` command is used to specify your command mode, and determines how subsequent commands entered from a terminal will be handled in the event of incomplete or incorrect parameters. The command has the following format:

`SET_COMMAND_MODE, SETCM`

mode, m : key normal, n, line, l, screen, s = \$required
status : var of status = \$optional

MODE, M: This parameter is used to specify one of the following command modes.

normal This is the mode under which all commands in procedures, batch jobs and included files are processed by default -- that is, if any error is detected during evaluation of the parameters for the command, processing of the command is terminated immediately.

line This mode is applicable only to interactive jobs. Line mode must be explicitly selected by the user via the SET_COMMAND_MODE command. This mode does not require any special type of terminal but does take into account certain terminal attributes such as page width. In line mode, the parameters entered with the command name are evaluated and for each detectable error, an error message is written and the user is prompted to enter the correction. The prompting information is derived from the data in the command's parameter description table (PDT).

Once all specified parameters have been accepted, a check is made to insure that all required parameters have been supplied and if not, the user is prompted to enter them.

When entering corrections or omitted parameters, the user may enter more than the parameter being prompted for. This is done in exactly the same way as parameters are normally given to a command.

If the original call to the command did not include any parameters and the command has at least one required parameter, the user is prompted for each parameter of the command, except those with the VAR attribute that are optional (this includes the STATUS parameter).

In response to any prompt, a terminate break can be entered to terminate processing of the command. Also, a question mark (?) can be entered to obtain information about acceptable responses to the prompt. If a question mark is entered twice in succession in response to a prompt, or a double question mark (??) is entered, information about how to use line mode facilities is displayed. A carriage return may be entered to skip the parameters, or to use the default value if provided.

34.3 Manual Errata

The following items are Manual Errata in the area of SCL:

1. Sample Prolog Is in Error
2. \$STRREP Can Convert Any Value to a String
3. SET_COMMAND_LIST Has PLACEMENT Parameter
4. Submitting Jobs from NOS
5. Submitting Jobs from NOS/BE

6. SCL Quick Reference Omissions
7. DISPLAY_FUNCTION_INFORMATION Command
8. NOS/VE Welcome Banner
9. \$PROCESSOR Function Produces Different Results

34.3.1 Sample Prolog Is in Error

The sample prolog shown on page 2-20 of the SCL System Interface Manual (60464014) should be replaced with the following:

```
SET_COMMAND_LIST a=.rmd.command_library
SET_PROGRAM_ATTRIBUTES tel=fatal
CREATE_VARIABLE n=response k=string
ACCEPT_LINE v=response i=input p='Choose 1)TTY, 2)Z150, 3)CDC721 '
IF response = '3' THEN
    SET_TERMINAL_ATTRIBUTES e=no pl=30 hp=yes pw=132
ELSEIF response = '2' THEN
    SET_TERMINAL_ATTRIBUTES e=no pl=24 hp=no pw=80 ..
    terminal_model=pc_connect_10
IFEND
DELETE_VARIABLE response
SET_WORKING_CATALOG $user
"End prolog processing."
```

34.3.2 \$STRREP Can Convert Any Value to a String

The documentation for the \$STRREP function in the SCL Language Definition Manual (60464013) doesn't indicate that this function can convert any type of value to a string. That is, in addition to converting an integer or boolean value, \$STRREP can convert a file, name, real, status, string, or variable value to a string.

The following sample procedure demonstrates usage of \$STRREP. No matter what type of value is specified for SUBJECT, it will be converted to a string.

```
PROC exp, e (
    subject : any = 'SCL SUBCOMMANDS LISTED BY UTILITY'
    manual  : file = $system.manuals.scl
    status  : var of status = $optional
)

    explain s=$strrep($value(subject)) m=$value(manual)

PROCEND exp
```

For example, executing this procedure by entering:

```
exp accl
```

would result in the procedure executing the command:

```
explain s='ACCL' m=$system.manuals.scl
```

34.3.3 SET_COMMAND_LIST Has PLACEMENT Parameter

The SCL Language Definition Manual (60464013) doesn't document the PLACEMENT (or P) parameter of the SET_COMMAND_LIST command. This parameter positionally follows the SEARCH_MODE parameter. You can specify the keywords BEFORE (or B) or AFTER (or A) for the PLACEMENT parameter. This parameter determines whether the entries to be added to the command list are placed before or after the current entries. Omission causes a value of BEFORE to be used.

34.3.4 Submitting Jobs from NOS

Support for the command

```
ROUTE,1fn,DC=LP,FC=RH.
```

was discontinued with NOS/VE 1.1.3 L644. Now, a user may submit NOS/VE jobs from NOS and the output will be sent back to where the job originated (i.e. remote batch terminal (RBF), another NOS mainframe, central site mainframe). The output for a NOS/VE batch job originated from NOS may also be sent to the NOS wait queue or suppressed altogether.

Examples:

```
ROUTE,nve-job,DC=IN,ST=nve-lid.
```

The output will be sent to the central site or the mainframe which the job was routed from.

```
ROUTE,nve-job,DC=TO,ST=nve-lid.
```

The output will be put in the NOS wait queue.

```
ROUTE,nve-job,DC=NO,ST=nve-lid.
```

The output will be suppressed.

```
ROUTE,nve-job,DC=IN,ST=nve-lid,UN=MDG.
```

The output will be sent to the RBF terminal that is logged in under user name MDG.

where "nve-lid" is a logical identifier (LID) defined for a user to route a job to NOS/VE and "nve-job" is the NOS file on which the NOS/VE job is stored.

Note that if you create your file through the CCL .DATA statement, be careful that underscores in your VE commands are kept intact. You can keep underscores from being removed by using the CCL

.EXPAND,OFF.

directive, or by changing the CCL concatenation character to an & with the .CC(&) directive, or by using a # (number sign) in front of each underscore, as in the following example.

```
.PROC,VEJOB*I"Demonstrate NOS/VE job routed from NOS"
.
$ROUTE,JOB,DC=IN,ST=NVE.
.DATA,JOB.
login u=myuser pw=mypassword fn=myfamily
display#_catalog c=$user
logout
```

34.3.5 Submitting Jobs from NOS/BE

As of NOS/VE 1.1.3 L644, a user is able to route jobs from NOS/BE to NOS/VE with either of the following statements:

```
ROUTE,lfn,DC=IN,ST=MFA.
ROUTE,lfn,DC=IN,ST=MFB,TID=nn.
```

where MFA and MFB are valid logical identifiers (LIDs) defined for NOS/VE.

In the first case, the output gets routed back to the mainframe it was sent from. In the second case, the output gets routed to the INTERCOM terminal specified by the TID parameter on the mainframe the job was sent from.

NOTE

It is not possible with the current version of RHF on NOS/BE to route a file to NOS/VE on other than the host mainframe.

34.3.6 SCL Quick Reference Omissions

The following functions are not documented in the SCL Quick Reference Manual (60464018):

\$job_output \$job_status \$output_status

These functions are described in detail in the SCL Language Definition Manual (60464013).

34.3.7 DISPLAY_FUNCTION_INFORMATION Command

This command displays information about a function and its parameters. The information includes the parameter types and default values. The information is derived directly from the function's parameter description table.

This command is not intended to be a replacement for information in on-line manuals, but rather a 'memory jogger' for any function. It will work for any function that could be called at the point where this command is called (including system supplied functions and utility supplied functions).

This command can be used to obtain information about the parameters for any function that uses SCL services to parse its parameters.

```
DISPLAY_FUNCTION_INFORMATION, DISFI  
    function, f: name = $required  
    output, o: file = $output  
    status
```

function, f This parameter specifies the function for which information is sought.

output, o This parameter specifies the file to which the parameter information is written.

Omission will cause \$OUTPUT to be used.

34.3.8 NOS/VE Welcome Banner

The NOS/VE welcome banner presented to a NOS/VE interactive user has been changed. Rather than display the model it now displays the performance class

of the machine. Formely, the welcome banner was:

```
Welcome to the NOS/VE Software System.  
Copyright Control Data 1983, 1986.  
CYBER 180-855 SN109,209.  NOS/VE 1.2.1 L664  
October 3, 1986.      4:27 PM.
```

The welcome banner now is:

```
Welcome to the NOS/VE Software System.  
Copyright Control Data 1983, 1986.  
CYBER 855 Class SN109,209.  NOS/VE 1.2.1 L670  
November 3, 1986.      4:27 PM.
```

In the new form, the performance class of the mainframe is displayed.

34.3.9 \$PROCESSOR Function Produces Different Results

The \$PROCESSOR function will now produce slightly different result for the parameters MODEL, M, MODEL_TYPE, and MT. The MODEL_TYPE parameter returns the class of the machine, as in:

```
CYBER 855 Class
```

instead of the former:

```
CYBER 180
```

The MODEL parameter is still defined to be

```
MODEL_TYPE/'-'//MODEL_NUMBER
```

as in:

```
CYBER 855 Class-855
```

Note that formerly it would have been:

```
CYBER 180-855
```

Chapter 35 System Performance and Maintenance

35.1 Incompatibilities

The following items describe incompatibilities in the analysis area:

1. Analyze Dump Restart Files Are Incompatible

35.1.1 Analyze Dump Restart Files Are Incompatible

Restart files saved on disk are not compatible with the NOS/VE 1.2.1 L670 Analyze Dump utility and the tapes should be re-read.

35.2 Significant Problems

The following items describe significant problems in the analysis area:

1. DISPLAY_BINARY_LOG Cannot Always Access an Active Log

35.2.1 DISPLAY_BINARY_LOG Cannot Always Access an Active Log

An initial deadstart or an upgrade deadstart will leave a local copy of the CYBIL runtime library attached to the system job. This copy of the CYBIL runtime library will not have the correct ring attributes for DISPLAY_BINARY_LOG to execute at ring 6 and, therefore, DISPLAY_BINARY_LOG cannot access an active global log.

Under these conditions DISPLAY_BINARY_LOG will abort with unsatisfied external references.

To correct this problem, enter the following commands at the NOS/VE console:

```
DETACH_FILE $local.cyf$run_time_library  
ATTACH_FILE $system.cybil.cyf$run_time_library
```

This will attach the CYBIL runtime library with the correct ring attributes.

PSR NV0J369

35.3 Notes and Cautions

The following items describe notes and cautions in the analysis area:

1. MAX_WORKING_SET Attribute
2. Online Manual of Site Analyst Examples
3. DISPLAY_SYSTEM_DATA Performance Tool

35.3.1 MAX_WORKING_SET Attribute

You should not set the MAX_WORKING_SET attribute for job classes (set by the CREATE_JOB_CLASS or CHANGE_CLASS_ATTRIBUTES commands) to greater than the number of pages available to users after deadstart.

The system defaults for job classes are adequate for systems with 14 MB or more of memory available to NOS/VE. For systems with less than 14 MB of memory, you must determine the number of pages available to users by doing a

DISPLAY_SYSTEM_DATA JMS

and adding the number of pages in the free and available queues (do this after the system has deadstarted, but before any jobs have logged in). Set the MAX_WORKING_SET default for each job class to be less than the number of pages in the free and available queues. The DISPLAY_SYSTEM_DATA command is documented in Appendix D of this SRB.

35.3.2 Online Manual of Site Analyst Examples

An online manual of site analyst examples is available. This manual works just like the online EXAMPLES manual (described in an appendix in this SRB). The examples in the site analyst manual illustrate installation, configuration, permanent file maintenance, and network management topics.

NOTE

The site analyst examples manual is not supported with NOS/VE 1.2.1 L670, but is provided for site analyst use and experimentation. Feedback is welcome through Central Software Support, but formal PSRs will not be answered.

35.3.3 DISPLAY_SYSTEM_DATA Performance Tool

Appendix D of the SRB describes the DISPLAY_SYSTEM_DATA (DISSD) command, which provides data for NOS/VE performance analysis.

Chapter 36 Terminal and Screen Management

36.1 Significant Problems

The following items are significant problems in the area of Terminal and Screen Management:

1. Dual State Transparent Mode Problem
2. NAM Login May Require Additional Carriage Return
3. Errors in Setting NAM/CDCNET Attributes
4. Error Message for Missing `TERMINAL_MODEL`
5. Caution Regarding Defining Terminals with Field Protect
6. Incomplete `AMP$PUT_PARTIAL` Is Not Flushed
7. Limited Range for Delay Parameters
8. XMODEM Needs Null `ATTENTION_CHARACTER` for NAM/CDCNET
9. MacConnect File Transfers
10. CDC721 Problems at 19200 BPS Using NAM/VE
11. Disconnect from 2550, Reconnect through CDCNET

36.1.1 Dual State Transparent Mode Problem

When the values for the connection attributes `Transparent_Character_Mode`, `Transparent_Length_Mode` and `Transparent_Timeout_Mode` mix any forwarding option (`IFC$TRANS_CHAR_FORWARD`, `IFC$TRANS_CHAR_FWD_TERMINATE`, `IFC$TRANS_LEN_FORWARD`, `IFC$TRANS_LEN_FORWARD_EXACT`, `IFC$TRANS_TIMEOUT_FORWARD`) with any termination option (`IFC$TRANS_CHAR_TERMINATE`, `IFC$TRANS_TIMEOUT_TERMINATE`, `IFC$TRANS_LEN_TERMINATE`) on dual state 2550 connections, single message transparency may be used instead of the appropriate multiple message transparency. This may cause unpredictable results.

PSR NV0J612

36.1.2 NAM Login May Require Additional Carriage Return

When logging in to NOS/VE through NAM, the clear screen request in the 'Welcome to NOS/VE' banner will place the terminal in page wait if the terminal has page wait enabled. This then requires the user to enter an additional carriage return after supplying his password.

PSR NV01896, NV0I565

36.1.3 Errors in Setting NAM/CDCNET Attributes

The NOS/VE `CHANGE_TERMINAL_ATTRIBUTES` (CHATA) or `SET_TERMINAL_ATTRIBUTES` (SETTA) commands will incorrectly set certain terminal attributes for a terminal connected to NAM/CDCNET.

The `CARRIAGE_RETURN_DELAY` and `LINE_FEED_DELAY` parameters yield CDCNET values much smaller than requested. The user can workaroud this error by using the CDCNET `%CHATA` command.

The `TERMINAL_MODEL` parameter is correctly changed within the NOS/VE environment, but CDCNET will not receive the change. Since this parameter is needed only for the NOS/VE screen products, which will work correctly, this is a cosmetic problem and no workaround is needed.

The `FOLD_LINE` parameter will have its boolean value reversed. The workaround is to use the opposite value of what is intended for the NOS/VE commands, or to bypass the NOS/VE commands and manipulate this parameter with the CDCNET `%CHATA` command.

PSR NV0J535

36.1.4 Error Message for Missing `TERMINAL_MODEL`

Usage of screen-mode products, such as `EDIT_CATALOG`, requires that the user has established a value for the `TERMINAL_MODEL` terminal attribute. Such products can return the error message:

```
--ERROR-- Improper Name
```

if the terminal model has not been defined.

PSR NV01827

36.1.5 Caution Regarding Defining Terminals with Field Protect

Users should note that when defining new types of terminals, a field protect capability should only be enabled if its behavior is reasonably close to that of the CDC721 and the `PC_CONNECT` emulator. The basic requirements are that protection not cause auto-tabbing, and that scrolling must be bounded at the first row which contains any protected characters. A terminal is allowed to transmit overstrikes of text to the host even if it has protected such overstrike from printing onto a protected location; the NOS/VE screen products will detect this discrepancy and ignore the keystroke. When the host paints the screen, it will send commands before and after the protected

text, expecting the terminal to protect all locations to which that text is printed. The host also sends out commands, before and after a complete block of output, to disable and then re-enable all protection.

36.1.6 Incomplete AMP\$PUT_PARTIAL Is Not Flushed

If a program has built part of a record of terminal output, via AMP\$PUT_PARTIAL, that text will not be output to the terminal by calling AMP\$FLUSH. The record must be first completed. Note that calling AMP\$GET_NEXT with a non-null prompt string attribute will successfully terminate the incomplete output.

PSR NVOI738

36.1.7 Limited Range for Delay Parameters

The terminal attributes CARRIAGE_RETURN_DELAY, FORM_FEED_DELAY, and LINE_FEED_DELAY are only allowed values up to 999 in the NOS/VE CHANGE_TERMINAL_ATTRIBUTES command. CDCNET allows values of up to 1000 for CRD and LFD, and 3000 for FFD. These parameters may be set via the CDCNET command %CHANGE_TERMINAL_ATTRIBUTES.

PSR NVOI089

36.1.8 XMODEM Needs Null ATTENTION_CHARACTER for NAM/CDCNET

Users must not have an ATTENTION_CHARACTER defined while using the XMODEM file transfer utility, if connected to NAM/CDCNET. For this type of connection, the only way to nullify the ATTENTION_CHARACTER is to use the CDCNET command:

%CHANGE_TERMINAL_ATTRIBUTES attention_character=0

Note that the problem does not apply for connections to NAMVE/CDCNET.

PSR NVOI823

36.1.9 MacConnect File Transfers

Version 1.0+ of MacConnect can transfer files only when the SCL prompt string is a slash. This causes problems when the current prompt is from a utility, such as SOURCE_CODE_UTILITY, EDIT_FILE, CREATE_OBJECT_LIBRARY, or TASK/TASKEND. The user can work around this problem by transferring files outside of such utilities, or by converting the utility prompt to a slash by performing the file transfer between the two commands:

```
INCLUDE_FILE input ''  
--file transfer--  
EXIT
```

Note that this only applies to Version 1.0+. It does not apply to Version 1.0 nor the next release of MacConnect, Version 1.1.

PSR NV0J421

36.1.10 CDC721 Problems at 19200 BPS Using NAM/VE

The CDC721 terminal is not able to reliably receive data at a line speed of 19200 bits per second. Text may be lost under a variety of conditions, including both NOS and NOS/VE and both line and screen modes, depending on the ROM version in the particular terminal. The problems are most severe for NOS/VE full screen applications when using 132 column format with quick response times from the host. For a more detailed discussion of this problem, refer to the "Flow Control" topic in the "CDCNET Operational Considerations" section of this SRB.

36.1.11 Disconnect from 2550, Reconnect through CDCNET

NOS/VE will allow mixed detaches and attaches for CCP and CDCNET connections, provided that the CDCNET connection uses NAM and dual-state access to NOS/VE. However, this is not a designed capability of NOS/VE, and results may be unpredictable. NOS/VE does not allow attachment of a detached job when that would mix NAM/VE with either type of dual-state connection.

PSR NV0I148

36.2 Notes and Cautions

The following items are notes and cautions in the area of Terminal and Screen Management:

1. Response After Terminate Break Interrupt
2. PAGE_WIDTH of Zero
3. Terminal Definition of Visual Attributes Restriction
4. Processing of Attributes Relevant to Only One Network
5. Interference between NOS/VE and CDCNET Commands
6. Full-Screen Products Now Support Auto-Tabbing
7. Caution to Users of Zenith Z29 Terminals
8. Terminal Interrupts in Screen-Oriented Products
9. Usage of EDIT_FILE on INTERCOM
10. XMODEM EC Parameter Usage

36.2.1 Response After Terminate Break Interrupt

When output is terminated with a terminate break (CONTROL/T for CCP or %2 for CDCNET), CDCNET ends the output at the end of a buffer, so CDCNET will print some additional text before accepting the interrupt.

36.2.2 PAGE_WIDTH of Zero

Setting the PAGE_WIDTH terminal attribute to zero can cause some display commands to format the output as though the page width is 34.

36.2.3 Terminal Definition of Visual Attributes Restriction

The current NOS/VE screen products do not support terminals that must consume a character on the screen in order to change visual attributes.

36.2.4 Processing of Attributes Relevant to Only One Network

Several of the terminal attributes and connection attributes are relevant for only a subset of the network subsystems supported by NOS/VE. For example, the PAUSE_BREAK_CHARACTER and TERMINATE_BREAK_CHARACTER are meaningful only for a 2550 running CCP with NAM and dual-state access to NOS/VE, and the ATTENTION_CHARACTER is only meaningful for CDCNET.

NOS/VE 1.2.1 L670 allows all parameters to be used in the commands to change attributes. When a user selects a parameter which is not available, the parameter is ignored with no warning message. The display commands show only those attributes that actually exist for the network currently in use.

36.2.5 Interference between NOS/VE and CDCNET Commands

NOS/VE and CDCNET both provide commands to alter terminal attributes and connection attributes. In some cases, there is no practical difference which command is used to change an attribute, but there are other cases where NOS/VE considers itself to "own" an attribute, and confusing results can occur if a CDCNET command changes that attribute without NOS/VE involvement. Since most users should not be concerned with these distinctions, it is suggested that the user should choose a NOS/VE command rather than the corresponding CDCNET command, and should use the CDCNET commands only while not logged in to NOS/VE or in special situations where the NOS/VE command cannot achieve the desired effect.

NOTE

This document purposely shows several examples where a CDCNET command must be used in place of a NOS/VE command.

One case where the user must use a CDCNET command, rather than NOS/VE, is to establish interrupt capability for screen-mode products for a terminal connected to CDCNET and NAM with dual-state access to NOS/VE. This situation is described in the 'Terminal Interrupts in Screen-Oriented Products' section of this chapter.

For terminals connected through CDCNET to NOS and dual-state NOS/VE, NOS/VE keeps its own record of terminal attributes and downloads changes to CDCNET. If an attribute is changed with a CDCNET command, then CDCNET and NOS/VE will have conflicting values and the results will be unpredictable.

For terminals connected to NAMVE/CDCNET, the CDCNET %CHANGE_TERMINAL_ATTRIBUTES command can be safely mixed with the NOS/VE SET_TERMINAL_ATTRIBUTES and CHANGE_TERMINAL_ATTRIBUTES commands. This is safe because NAM/VE uses CDCNET to store the current attributes. However, the CDCNET %CHANGE_CONNECTION_ATTRIBUTES command should not be used in any normal situation, as NOS/VE keeps its own connection attributes for each instance of open for each terminal file, and downloads changes to CDCNET incrementally when there are differences in the attributes associated with the files. Any attribute change introduced by the CDCNET %CHANGE_CONNECTION_ATTRIBUTES command will have an unpredictable lifespan.

The NOS/VE CHANGE_TERMINAL_ATTRIBUTES command allows the user to specify all parameters that can be supported by NAMVE/CDCNET, NAM/CDCNET, NAM/CCP, and INTERCOM. For each network, only a subset of the terminal attributes is actually manipulated. The CYBIL File Management Manual (60464114) details the incompatibilities.

36.2.6 Full-Screen Products Now Support Auto-Tabbing

Full screen products such as EDIT_FILE, Programming Environment, IM/Quick, and EDIT_CATALOG, now support the "auto-tabbing" feature that is a characteristic of some terminals. Auto-tabbing is defined as a behavior where the terminal automatically jumps the cursor forward to the next unprotected field, when the user has typed text into all characters of the current unprotected field.

Of the terminals pre-defined by CDC, three have auto-tabbing capabilities: CDC721, PC_CONNECT_11, and CDC722_30. The 721 and Connect can enable or disable the auto-tab option independently of the enabling of field protection, while the 722_30 always autotabs if protection is in use. The 721 and Connect have an added behavior with autotabbing: they will not allow the left and right cursor keys to move into a protected area. This behavior can cause certain products (including Screen Design Facility) to be inherently unusable; therefore the auto-tabbing option is disabled by default for CDC721 and PC_CONNECT_11. These two terminal definitions now include embedded comments that show a user how to turn auto-tabbing back on, for the benefit of users who want to conserve keystrokes for some screen products and do not intend to use SDF.

Because the 722-30 does not restrict cursor motion, auto-tabbing is used by default. Previous definitions of the 722-30 did not use protection; the Level 670 definition enables protection as the means to achieve auto-tabbing. With the 722-30, protection is associated with bright or dim video attributes. Three configurations are possible:

- . All fields bright, no protection or autotabbing.
- . Protected areas bright and input areas dim, with autotabbing.
- . Input areas bright and protected fields dim, with autotabbing.

The choice of "best" configuration for the 722_30 depends on each user's work habits: The second configuration looks best for products that protect most of the screen area and have "fill in the forms" orientation. The third configuration looks best for products that use most of the screen for input fields and encourage general editing with insertions and deletions. For a user who frequently flips between both usage styles, the first configuration may be the best compromise.

CDC has chosen the third configuration as the default. Note that the second configuration has a number of side-effects when line insertion is performed in the full screen editor, as the newly inserted blank line will initially be protected by the hardware, even though it should not be; the editor will detect this discrepancy and repaint the affected areas.

The CDC722_30 terminal definition contains embedded comments that show users how modify the 722_30 protection configuration when CDC's default is not optimal.

The CDC722_30 terminal (not to be confused with the older CDC722) will experience auto-tabbing problems if protected fields are used, as the CDC722_30 always auto-tabs. The CDC-released terminal definition for the CDC722_30 will avoid this problem by not using protected fields. Users are advised to NOT enable protected fields with any modification to the CDC722_30 terminal definition.

Code is being developed to support autotabbing on the CDC721, PC_CONNECT_11, and CDC722_30 terminal models. This code will not be available in the initial release of 1.2.1 but is expected to appear in a BCU shortly after the release. This code will also have the effect of allowing protected fields to be used correctly for the first time for terminals that associate protection with one of the visual attributes, e.g., low-intensity or underlining or inversion. This is characteristic of many ANSI X3.64 protocol devices.

36.2.7 Caution to Users of Zenith Z29 Terminals

Full screen products such as EDIT_FILE, Programming Environment, IM/Quick, and EDIT_CATALOG, support the Zenith Z29 slightly differently than the Zenith Z19, in terms of how the hardware modes are set when the screen application finishes and sets the terminal for line mode usage. The Z19 is set to its

power-on defaults, which the user could have previously set to his or her favorite values by means of switches located inside the terminal housing. The Z29 is not set to its power-on defaults because some types of modems will drop their phone connection when the Z29 resets itself.

Instead, the screen applications will set the Z29 to the following modes:

- Unshifted Keypad
- Cursor Wraparound at End of Line
- Hold Screen Key Disabled

Some users may prefer a different set of hardware modes. These users can use the `DEFINE_TERMINAL` command to create their own definition of the Z29. This particular situation is controlled by the "set_line_mode" directive of the terminal definition language. The user can reverse any of these three modes, or if the modem hangup is not considered important, the user can use the Z19 technique. Currently, including hold screen key enabled in the set_line_mode sequence causes Z29's to hang under some circumstances. For additional information, refer to the NOS/VE Terminal Definition Manual (60464016).

36.2.8 Terminal Interrupts in Screen-Oriented Products

NOS/VE 1.2.1 L670 is the first release to fully support pause break and terminate break interrupts while in screen mode. Previous releases had limited screen-mode support for these interrupts.

To successfully use this capability, the user must be aware of which of the network configurations is being used, and must set certain attributes before trying to use the interrupts in screen mode.

Configuration 1: 2550 (CCP) communicating thru NAM into the VEIAF (Dual-State) subsystem.

The user will use the same interrupt keystrokes in screen mode as in line mode. By default, these keystrokes are CONTROL/T plus carriage return for terminate break, and CONTROL/P plus carriage return for pause break. For CCP to treat these keystrokes as interrupts in screen mode, the user must have previously used the NOS/VE command:

```
CHANGE_TERMINAL_ATTRIBUTES attention_character=char-value
```

where "char-value" is any non-null character value. It does not matter what this character value is, so long as it is not `$CHAR(0)`. The user does not use this character for the interrupt, the user uses the regular terminate or pause break characters, which default to CONTROL/T and CONTROL/P plus carriage return.

Note that if `ATTENTION_CHARACTER` remains as `$CHAR(0)`, which is the default, then the CONTROL/T and CONTROL/P characters are text characters in screen mode rather than interrupts. This could be valuable for terminals that use these control codes for screen manipulations. For such terminals, the user

could also use the `TERMINATE_BREAK_CHARACTER` and `PAUSE_BREAK_CHARACTER` attributes to select alternate keystrokes for interrupts.

Configuration 2: DI (CDCNET) communicating thru NAM into the VEIAF (Dual-State) subsystem.

The user will not use the terminate break or pause break keystrokes (%2 or %1 plus carriage return) to interrupt while in screen mode. Instead, the user will use the `ATTENTION_CHARACTER` keystroke, without a carriage return. The user must first define an `ATTENTION_CHARACTER`, and must set the `ATTENTION_CHARACTER_ACTION` to 1 if pause break behavior is desired, or to 2 if terminate break behavior is desired.

NOS/VE does not have commands or program interfaces to set the attributes for this communication configuration. The user should use the CDCNET commands to set the desired attributes, as in this example:

```
%CHANGE_TERMINAL_ATTRIBUTES attention_character=char-value
%CHANGE_CONNECTION_ATTRIBUTES attention_character_action=1 or 2
```

where "char-value" is the desired character, in CDCNET syntax. For ASCII control characters, you can use the integer value, such as 3 for the CONTROL/C key.

Configuration 3: DI (CDCNET) communicating thru NAM/VE directly to NOS/VE, without NOS Dual State. This is the only configuration present in a machine deadstarted in standalone mode.

As with configuration 2, the user will use the `ATTENTION_CHARACTER` keystroke to interrupt screen mode rather than the terminate break or pause break sequences. The `ATTENTION_CHARACTER` and `ATTENTION_CHARACTER_ACTION` must be setup in advance for this to be possible. However, the user should use NOS/VE commands to initialize the attributes rather than CDCNET commands:

```
CHANGE_TERMINAL_ATTRIBUTES attention_character=char-value
CHANGE_TERM_CONN_DEFAULTS attention_character_action=1 or 2
CHANGE_TERM_CONN_ATTRIBUTES terminal_file_name=Input ACA=1 or 2
CHANGE_TERM_CONN_ATTRIBUTES terminal_file_name=Output ACA=1 or 2
CHANGE_TERM_CONN_ATTRIBUTES terminal_file_name=Command ACA=1 or 2
```

36.2.9 Usage of `EDIT_FILE` on INTERCOM

The NOS/BE terminal driver always issues a line feed (LF) in response to a carriage return (CR) sent from a terminal. This causes a problem when using `EDIT_FILE` with terminals which consider the home position (command line) to be the last line of the display and do not have a page mode. For such terminals (e.g., the 722-10) the user should clear the screen (followed by a CR) to cause `EDIT_FILE` to repaint the entire screen.

36.2.10 XMODEM EC Parameter Usage

If your micro file transfer application uses only checksum error checking (e.g. CROSSTALK) then you must specify EC=CHECKSUM on your XMODEM command.

36.3 Manual Errata

The following are manual errata in the Terminal and Screen Management area:

1. NAM/CCP Terminal Attribute Mapping Error

36.3.1 NAM/CCP Terminal Attribute Mapping Error

The NAM/CCP Mapping table in Appendix F of the CYBIL File Management Manual (60464114D) has an error in it. On pages F-6 and F-7, the FN/FV pair (46,0) should have the footnote: If any other terminal attributes cause the (46,1) pair to be set, the (46,1) will take precedence over the (46,0). (46,1) is multiple-message transparent mode.

Chapter 37 Validation

37.1 Notes and Cautions

The following items describe validation notes and cautions:

1. \$SYSTEM Username's Validations and Security
2. Preventing Interactive Access by a User

37.1.1 \$SYSTEM Username's Validations and Security

When an initial deadstart is performed, NOS/VE creates the \$SYSTEM username with a password of \$SYSTEM and validates it for all job classes.

Sites should, at a minimum, change the password for the \$SYSTEM username to prevent unauthorized users from logging into \$SYSTEM. The password change can be accomplished by using either SET_PASSWORD or ADMINISTER_USER from the system console.

Sites may also want to prevent interactive access (via NAM/VE) to \$SYSTEM by setting the default interactive job class to NONE and removing all interactive job classes from the list of valid job classes for \$SYSTEM.

It is important to remember that \$SYSTEM is a username just like other usernames as far as the validation process is concerned. To insure adequate protection, the password for \$SYSTEM should be changed regularly.

37.1.2 Preventing Interactive Access by a User

Sites may want to prevent interactive access (via NAM/VE) to a user by setting the default interactive job class to NONE and removing all interactive job classes from the list of valid job classes for the user. The TIMESHARING capability is not used to control access in this release.

37.2 Manual Errata

The following are manual errata in the validation area:

1. NOS and NOS/VE Password/Validation Relationship Incorrect
2. Order of Epilog Execution Incorrect

37.2.1 NOS and NOS/VE Password/Validation Relationship Incorrect

Chapter 1 of the Family Administration Manual (60464513) incorrectly documents the relationship between NOS dual state and NOS/VE user validation. In particular, it is not true that the NOS interactive password must be identical to the NOS/VE password. The following describes the relationship between NOS dual state and NOS/VE user validation for users who access NOS/VE through NAM.

1. For all users, the NOS and NOS/VE user names must be identical. NOS/VE user names must be SCL names (i.e., the first character cannot be numeric).
2. For users who need to use interstate communications, either their NOS/VE password must be identical to their NOS batch password or their link attributes must specify their correct NOS batch validation. These users must also be validated for the NOS permissions CUCP and CNVE. Interstate communications include the following:
 - a. GET_FILE or REPLACE_FILE commands
 - b. PRINT_FILE command specifying DUAL_STATE_ROUTE_PARAMETERS
 - c. CREATE_INTERSTATE_CONNECTION command
 - d. File Migration Aid (FMA)
 - e. Interstate communication as described in the CYBIL System Interface Manual (60464115).

Additionally, users who need to route jobs from NOS to NOS/VE must also be validated for the NOS permissions CUST and CQLK.

37.2.2 Order of Epilog Execution Incorrect

On page 1-14 of the Family Administration Manual (60464513) the sentence:

The user epilog executes after the system and job class epilogs, if any.

Should be replaced by:

The user epilog executes before the job and system class epilogs, if any.

Appendix A

Release Documentation Bulletins

This appendix describes the Control Data documentation bulletins that are associated with a NOS/VE release. Both printed and machine readable copies of all the bulletins (except as noted) are included with your release materials.

A.1 Software Release Bulletin (SRB)

The SRB contains information about installation and usage of the new release pertinent to both site analysts and users.

Following the installation documentation, the SRB is organized alphabetically into chapters on different areas of the operating system and products. Each chapter is divided into the topics (if applicable) of: installation/upgrade, incompatibilities, significant problems, notes and cautions, manual errata, and future considerations.

Incompatibility information for products affecting more than just the site analyst is described in the User Impact Bulletin.

A.2 User Impact Bulletin (UIB)

The UIB provides incompatibility information about NOS/VE products. The audience ranges from an applications analyst to an end user.

The UIB is organized into chapters that parallel the products discussed. For example, there may be a chapter on IM/DM, another on CYBIL, and so on.

The UIB is distributed only when there are incompatibilities in a release.

Any incompatibility information concerning a site analyst is listed in the SRB.

A.3 Software Availability Bulletin (SAB)

The SAB describes the feature content of a NOS/VE release.

The SAB is organized by products. Brief descriptions of the new features for each product are included with the manuals that are affected by the new feature.

The SAB also contains the following items:

1. Ordering information for all manuals and microfiche source listings associated with NOS/VE.
2. Configuration management data indicating mainframe certification levels and Field Change Announcements (FCA) indexes.
3. Support information describing the critical support and dual-state supported partners timetable.

The SAB is only available in printed format with your release materials.

A.4 Installation Bulletins (IB)

IBs are issued by Central Software Support to alert your site about significant software problems identified during the life of this NOS/VE release. They are available for online access through SOLVER.

Contact your local Control Data Professional Services for information on how to access SOLVER.

Since IBs are issued after the release, no printed or machine readable copies are included with the release.

A.5 Loading Machine Readable Copies

As mentioned previously, machine readable copies of some of the release documentation bulletins are included on the ONLINE MANUALS release tape (VSN=NVE007). When you install NOS/VE 1.2.1 L670, the bulletins are automatically loaded as the following permanent files:

SRB \$system.documentation.software_release_bulletin

Do the following if you wish to load these files on to a version of NOS/VE prior to NOS/VE 1.2.1 L670.

1. Create the following procedure on a permanent file:

```
PROC load_bulls (  
  bulletin, b          : key srb = $required  
  destination_file, df : file = $required  
  status               : var of status = $optional  
)  
  
  request_magnetic_tape f=$local.tape evsn='NVE007'  
  
  bull_name = $string($value(b))
```

```
IF bull_name = 'SRB' THEN
  bull_name = 'software_release_bulletin'
IFEND
RESTORE_PERMANENT_FILE l=output
  restore_file f=$fname(..
    $backup_file($local.tape, identifier) //'.'//bull_name) ..
    bf=$local.tape nfn=$value(df)
QUIT
detach_file f=$local.tape

PROCEND load_bulls
```

2. Execute this procedure as a job initiated from the operator console, specifying:

b The name of the bulletin you wish to load.
df The name of the permanent file copy of the bulletin. This must be a permanent file name.

The job will request the NVE007 tape. Mount this tape and assign it.

For example, if the procedure were on the file :NVE.ME.LB, the following commands entered from the operator console initiate a job to load the Software Release Bulletin to :NVE.ME.SRB:

```
job
:nve.me.lb b=srb df=:nve.me.srb
jobend
```

3. Create permanent file permits to the bulletins as appropriate.

Appendix B

Site Analyst Documentation Overview

This appendix lists the manuals that are pertinent to a site analyst. The major topic areas for each manual are also included. Ordering information for these and all other NOS/VE manuals is available in the Software Availability Bulletin.

NOS/VE System Performance and Maintenance Manual (60463915)

Documents NOS/VE system attributes, creating and changing job classes, the logical and physical configuration utilities, tailoring the system startup commands file, creating deadstart tapes and disks, the UPGRADE_SOFTWARE utility, system core debugger and dump analyzer commands, and the use of the NOS/VE statistics facility. The last chapter of the manual lists the contents of all NOS/VE statistics, including those that are emitted for disk and tape failures.

NOS/VE Operations Manual (60463914)

Describes each operator command, the parameters for the command, and how to interact with the operator console. Discusses the steps involved in deadstarting and terminating NOS/VE in either standalone or dual-state mode. Other sections of this manual document magnetic tape usage, the permanent file backup and restore utilities (including the reloading of permanent files), how to send messages to interactive users, and system step/resumption.

Family Administration Manual (60464513)

Contains the information needed to validate NOS/VE users. It also contains information for host (NOS or NOS/BE) validation of dual-state users.

CYBER Initialization Package(CIP) Manual (60457180)

Documents installation and use of the CYBER Initialization Package (CIP) components. These components are microcode, console drivers, boot programs for the deadstart of NOS/VE, and utilities for dumping memory.

NOS/VE Terminal Definition Manual (60464016)

Lists the commands used to define a terminal's capabilities for full screen usage and documents the process of creating (or updating) terminal definition files.

NOS/VE Network Interface Usage Manual (60463916)

Describes the Network Application Management Utility (MANAGE_NETWORK_APPLICATION) which is used to define and control network applications.

CDCNET Network Operator Utility Manual (60461520)

Describes the Network Operator Utility which is used to monitor, control, and dynamically reconfigure CDCNET.

NOS/VE Accounting Analysis Manual (60463923)

Written for the person responsible for customer accounting, this manual documents the Accounting Analysis System and how to create and maintain the pricing information used to bill customers.

CDCNET Site Administration and Configuration Manual (60461550)

Provides information and directions you need to configure the network software for CDCNET.

CDCNET Batch Device User Guide Manual (60463863)

Provides information and directions you need to define, configure, and use batch devices for CDCNET.

NOS Installation Manual (60459320)

Documents the installation and support for NOS and 170 portions of NOS dual-state systems. Includes instructions for rebuilding and applying BCUs for the 170 code used in the execution of dual state.

Appendix C

NOS/BE Dual-State Preparation

This chapter describes the steps you must perform to prepare the NOS/BE operating system for dual-state operations with NOS/VE 1.2.1. The chapter also describes the creation of NOS/BE procedures that initiate NOS/VE deadstarts. When you have completed the steps included in this chapter, refer to chapter 1 for instructions on installing NOS/VE.

When running NOS/VE in dual-state, the CYBER Initialization Package (CIP) must be installed on a device that is accessible to NOS/BE. Because CIP is accessed through NOS/BE, CIP cannot be installed on a device that belongs exclusively to NOS/VE.

C.1 Software Requirements

The minimum software requirements for NOS/BE Version 1.5 are sufficient to install NOS/VE. Use of the NOS/VE Interactive Facility (VEIAF) requires the installation of INTERCOM 5. The installation process assumes the presence of INTERCOM 5 and VEIAF; however, all interactive procedures can be executed in batch jobs.

C.2 Building NOS/VE Dual-State Binaries on NOS/BE

NOS/VE dual-state code can be rebuilt on NOS/BE by using the installed deck NVEBUILD from the DECKS program library. This deck will use as input the dual-state source library and any local code that the site has. This deck must be modified to change the ID's for the permanent file operations. See documentation included in the deck for further information.

The NVEBUILD deck assumes that the level of NOS/VE that you want to build is already installed on the system. NVEBUILD requires the use of files from that level of the NOS/VE installation.

The NVEBUILD deck does not rebuild the CYBIL part of NOS/VE dual-state, since CYBIL is a separate product and does not have to be installed for dual-state to operate. If you have installed CYBIL and you want to rebuild the CYBIL portions, the NVEBUILD deck can be modified to do so. Please see the documentation in the NVEBUILD deck for further details.

C.3 Step 1 Establish an Installation User Name

The first step in the installation process is to select a user name on NOS/BE under which you will install NOS/VE.

NOS/BE permanent files created during the NOS/VE installation or upgrade process are associated with a specific NOS/VE user name. All permanent files created during installation will have this user name as a permanent file ID. The user name selected in this step should be used whenever a UN parameter is specified in an installation procedure call. This user name will be validated in Step 7.

C.4 Step 2 Generate NOS/BE Configuration Changes

Steps 2 through 6 generate and install NOS/BE configuration changes required to install NOS/VE.

The process of creating NOS/BE deadstart files tailored for individual sites is described in the NOS/BE Installation Handbook. In addition to the material contained in that handbook, the following items should be addressed in preparation for the generation of a new NOS/BE deadstart file to support dual-state configuration:

1. NVE subsystem support
2. Equipment status table (EST) definitions
3. Central memory (CM) allocation

A. NVE Subsystem Support

The NOS/VE operating system appears to NOS/BE primarily as the NVE subsystem. In order to fully support the NVE subsystem, add the following to the CMR installation parameters (CMRIP):

*INSERT,CMRIP.1

SSNVE	CEQU	SS.NVE	(Turns on NVE subsystem support)
N.SBSYS	CEQU	5	(Sets the number of subsystems)

B. EST Definitions

Be sure that all equipment to be used by either NOS/BE or NOS/VE is defined in the equipment status table (EST). Define disk units to be used by NOS/VE as IDLE and disk channels to be used by NOS/VE as OFF.

C. CM Allocation

NOS/BE deadstart tapes may be prepared with or without Unified Extended Memory (UEM) defined. In cases where UEM is defined, its use is optional at deadstart time. To use UEM, however, a system must have more than 8 megabytes of central memory. To use UEM, NOS/BE needs 2 megabytes of central memory (not counting the memory assigned to UEM). In addition, NOS/VE needs 6 megabytes. Thus, an 8 megabyte system would have no

memory to spare for UEM.

Without UEM:

The amount of central memory allocated to NOS/BE is normally 2 megabytes (262,144 words), with the remainder assigned to NOS/VE. If that remainder is less than the minimum needed for NOS/VE, then the central memory allocated to NOS/BE can be automatically reduced to help increase the amount available to NOS/VE. For the parameter settings needed to make this automatic reduction, refer to chapter 1 of the NOS/BE Installation Handbook, under Deadstart Installation Parameters.

With UEM:

The amount of memory assigned to NOS/BE is the sum of the 2 megabytes (262,144 words) used as central memory and the amount defined as UEM by the Extended Memory Label which is constructed in CMR by the ECSLABEL and ECSPART macros, as described in the NOS/BE Installation Handbook. This amount can never be greater than a total of 16 megabytes (2,097,152 words). The remaining amount is assigned to NOS/VE. It must be at least 6 megabytes.

In the following example, assume the mainframe has 12 megabytes (1,572,864 words) of memory. The example constructs an ECS label which defines 2 megabytes (1000000B words) of memory as UEM for NOS/BE. A NOS/BE system deadstarted with this label would then require a total of 4 megabytes of memory, 2 megabytes for CM and 2 megabytes for UEM, leaving the remainder of the memory for NOS/VE. This amount would be slightly less than 8 megabytes, since some memory is used for the environment interface (EI) and the page table.

```
*INSERT CMR.2167
ECSLABEL      1000
ECSPART       NOSBE,1,300
ECSPART       NOSBE,2,477
```

The ECS label defines 1000000B words (262,144 decimal words) in this example, with 300000B words (98,304 decimal words) as a direct access partition and 477000B words (163,328 decimal words) as a paged partition; both are assigned to the mainframe with the ID NOSBE. The sum of the partitioned area is 1000B words (512 decimal words) less than the total amount of memory for UEM in order to accommodate the ECS label.

C.5 Step 3 Install the NOS/VE Release Files

To install the NOS/VE release files, you must run the build job NVEINST from the DECKS program library. This build job must be modified to use the ID defined in Step 1. The build job requires the NOS/BE dual-state tape for NOS/VE.

C.6 Step 4 Create the NOS/BE Deadstart File

You must now generate a new NOS/BE deadstart file containing the changes you made in Step 2 and the binaries and procedures from files NBEBINS and NVELIBB.

The NOS/BE Installation Handbook describes the process of building a NOS/BE deadstart tape. You should be familiar with this process before attempting this step. The standard NOS/BE installation decks have been modified to support installation of NOS/VE. These modifications include a new deck, NVEIE, to install the NOS/VE release files. The installation decks are contained on an UPDATE program library on file 3 of the NOS/BE batch corrective code (BCC) tape.

Submit the following job to obtain a listing of the installation decks and then save the DECKS old program library on a permanent file for later use.

```
DECKS,T100,PE1.  
REQUEST(OLDPL,PE)  
REQUEST(NEWPL,PF)  
SKIPF(OLDPL,2,17)  
UPDATE(F,N,*==,C=O,R,L=A7)  
CATALOG(NEWPL,DECKS,ID=INSTALL)  
6/7/8/9
```

You must generate an UPDATE modification set against the DECKS old program library (OLDPL) in order to create the NOS/BE build jobs. This modification set must include the following statement:

```
=DEFINE NOSVE
```

This statement will cause the build jobs that are generated to include NOS/VE in the list of products being installed. You must also include changes to the NVEIE job in the modification set. This replaces the username fields in the NVEIE job with the user name you defined in Step 1. Execution of the NVEIE build job is dependent on the NOS/VE release files being available as permanent files and cataloged with the user name from Step 1 as the ID.

C.7 Step 5 Redeadstart NOS/BE

Redeadstart NOS/BE using the deadstart file created in Step 4.

C.8 Step 6 Add Information for Partner Jobs

Steps 6 and 7 make NOS/BE accounting changes for the NOS/VE partner jobs. Step 6 is required only if your site has local accounting requirements; if not, go to Step 7.

Step 6 describes how to format the partner job template. The partner job template contains the accounting information your site has designed for NOS/BE. This accounting information is used by the NOS/VE Interim Remote Host Facility (IRHF) and Interstate Communication Facility (ICF) in creating NOS/BE job images. The partner job template contains a field that contains the NOS/BE job statement parameters and optional fields containing the NOS/VE family user accounting parameters. They must be in a format that can be processed by your site's NOS/BE batch job accounting utility. NOS/VE users have four accounting parameters: user name, password, account number, and project number (described in the NOS/VE Family Administration manual). These parameters may be specified in the NOS/BE partner job image according to the format of the partner job template. Examples of how to format the partner job template are at the end of this step.

The partner job template is a record that contains one or more of the following fields.

- &JOB This field will be replaced by the NOS/BE job statement parameters specified in the partner job template.
- &USER This field, if present, will be replaced by the NOS/VE user name.
- &PASSWORD This field, if present, will be replaced by the NOS/VE password.
- &CHARGE This field, if present, will be replaced by the NOS/VE account name.
- &PROJECT This field, if present, will be replaced by the NOS/VE project name.

The partner job template records are on the released procedures ACCFILE and RUNIRHF. ACCFILE and RUNIRHF are released on file NVELIBB and support the generation of a standard NOS/BE job control statement. The procedure ACCFILE is executed by the NOS/VE deadstart procedure from the NVELIB library during the deadstart process. When ACCFILE is executed, two partner job templates are created as local files named RHACCNT and ICACCNT. The RHACCNT file is used for all GET_FILE and REPLACE_FILE partner jobs. The ICACCNT file is used for all Interstate Communication partner jobs (like CREIC partner jobs). The RUNIRHF procedure is executed whenever the IRHF170 job is brought up. When RUNIRHF is executed, a partner job template is created as a local file named PRACCNT. The PRACCNT file is used for the PRINT_FILE command that uses the DSRP parameter. These partner job templates are then available to the IRHF and ICF when NOS/BE jobs are submitted.

To access and modify the released versions of ACCFILE and RUNIRHF, execute the following commands:

```
ATTACH,NVELIB,ID=username,MR=1.  
LIBRARY,NVELIB.  
GETPROC,ACCFILE,NVELIBB,UN=username.  
GETPROC,RUNIRHF,NVELIBB,UN=username.
```

where "username" is the user name you selected in step 1.

These commands get procedures ACCFILE and RUNIRHF from NVELIBB and copies them to local files named ACCFILE and RUNIRHF. You need to modify the partner job template records; in procedure ACCFILE, they are the RHACCNT and ICACCNT data records), in procedure RUNIRHF, it is the PRACCNT data record. Modify it to reflect your site's accounting requirements.

```
.PROC,ACCFILE*I,  
.  
.HELP  
THIS PROCEDURE CREATES THE PARTNER JOB TEMPLATE FILE USED  
BY THE REMOTE HOST AND INTERSTATE COMMUNICATION EXECUTIVES  
TO GENERATE PROPER ACCOUNTING INFORMATION FOR PARTNER JOBS  
THE CARDS IN THE ACCOUNT DATA SECTION MAKE UP THE PARTNER  
JOB FILE IMAGE.  
  
.ENDHELP  
REWIND,ICACCNT.  
REWIND,RHACCNT.  
REVERT. ICACCNT AND RHACCNT FILES CREATED.  
.DATA,ICACCNT.  
&JOB  
.DATA,RHACCNT.  
&JOB
```

ACCFILE and RUNIRHF can then be replaced on NVELIBB by the following commands:

```
REPPROC,ACCFILE,NVELIBB,UN=username.  
REPPROC,RUNIRHF,NVELIBB,UN=username.
```

where "username" is the user name you selected in Step 1.

NOTE

The CCL procedure RUNIRHF, which is on NVELIBB, is used to initiate the NOS/VE remote host job. If your site requires accounting information to be added to the remote host job record contained in RUNIRHF, you must do so before starting Step 7. To add the accounting information, enter the commands listed in Step 6 for accessing and modifying the release version of ACCFILE, substituting RUNIRHF for ACCFILE.

The following examples demonstrate how you might replace the DATA section of the ACCFILE procedure to describe the accounting parameters for your site.

Example 1:

If your site does not require batch job accounting parameters, no changes need to be made to the released version. It remains:

&JOB

Example 2:

If your site requires the user name, account number, and project number on the job control statement, replace &JOB with:

&JOB &USER,&CHARGE,&PROJECT

Example 3:

If your site requires the user name, password, account and project number as input to a local account program, replace &JOB with:

&JOB
ACCOUNT(&USER,&PASSWORD,&CHARGE,&PROJECT)

C.9 Step 7 Validate Users on NOS/BE

In order to be an interactive user of NOS/VE, you must be a validated user of NOS/BE INTERCOM 5. In order to log in interactively to NOS/VE, you must first log in to INTERCOM 5 on NOS/BE and then log in to NOS/VE by executing an INTERCOM 5 multiuser job named VEIAF. You must use the PASSWRD Utility (used to validate NOS/BE INTERCOM 5 users) to validate yourself as a NOS/VE user as well. A complete description of the PASSWRD Utility can be found in the INTERCOM 5 section of the NOS/BE Installation Handbook. The NOS/VE user name you established in Step 1 must be validated in the INTERCOM password file.

You can use the following job to validate a NOS/VE user:

JOB CARD.
PASSWRD.
7/8/9
OLD
ADD U=beuser,P=password,W=veuser
6/7/8/9

- U The NOS/BE user name of up to 10 alphanumeric characters. A user name must be specified for restricted passwords; it must be left blank or omitted for unrestricted passwords.
- P The NOS/BE password of up to 10 alphanumeric characters. A password is required.
- W The NOS/VE user name to be associated with the specified NOS/BE user name and password. It must be from 1 to 7 alphanumeric characters. The W parameter must be specified to permit this NOS/BE user to log in to NOS/VE.

Appendix D

Installation Commands

This chapter documents the following commands:

ACTIVATE_PTF	Activates a system task for the PTF server.
ACTIVATE_SCF	Activates a system task for the SCF application.
ACTIVATE_SCFS	Activates a system task for the SCF server.
DEFINE_PTF	Defines the PTF network applications.
DEFINE_SCF	Defines the SCF network applications.
DEFINE_SCFS	Defines the SCF server network applications.
SETUP_INSTALLATION_PROCESS	Creates a family, user name, and catalog permits for catalogs associated with installation.
SETVE	Creates the dual-state procedure file that initiates NOS/VE deadstart.
SET_VERSION_LEVEL	Changes the CDCNET software version level.

D.1 ACTIVATE_PTF Command

Purpose Defines and activates a system task for the PTF server.

Format ACTIVATE_PTF
LOAD_MAP=file
STATUS=status variable

Parameters LOAD_MAP or LM

Specifies a file to which the load map should be written. The default file name is \$NULL.

STATUS

Contains the completion status of this command.

Remarks The following remarks apply:

1. The name of the system task created by this command is OSASFILE_TRANSFER_SERVER. You will need to know this name for any subsequent deactivations or reactivations of the PTF server system task.
2. This command can be used only once per deadstart.

D.2 ACTIVATE_SCF Command

Purpose Defines and activates a system task for the SCF application.

Format ACTIVATE_SCF
LOAD_MAP=file
STATUS=status variable

Parameters LOAD_MAP or LM

Specifies a file to which the load map should be written. The default file name is \$NULL.

STATUS

Contains the completion status of this command.

Remarks The following remarks apply:

1. The name of the system task created by this command is SCF_CLIENT. You will need to know this name for any subsequent deactivations or reactivations of the PTF server system task.
2. This command can be used only once per deadstart.

D.3 ACTIVATE_SCFS Command

Purpose Defines and activates a system task for the SCF server.

Format ACTIVATE SCFS
APPLICATION_NAME=name
CONTROL_FACILITY_NAME=name
SYSTEM_TASK_NAME=name
LOAD_MAP=file
LOGGING = boolean
STATUS=status variable

Parameters APPLICATION_NAME or CFN

Specifies the name of the SCFS application (as defined previously on the DEFINE_SCF command). The default name is OSASSTATUS_CONTROL_FAC_SERVER. See the Remarks section on the following page for further restrictions on the use of this parameter.

CONTROL_FACILITY_NAME or CFN

Defines a name for the facility control. The default name is STATION_CONTROLLER_1. See the Remarks section on the following page for further restrictions on the use of this parameter.

SYSTEM_TASK_NAME or STN

Defines the name of the system task. The default name is SCF_SERVER. See the Remarks section on the following page for further restrictions on the use of this parameter.

LOAD_MAP or LM

Specifies the file to which the load map should be written. The default file name is \$NULL.

LOGGING or L

Specifies whether or not to log SCF server events; the default value is FALSE.

TRUE SCF server events are logged.
FALSE SCF server events are not logged.

STATUS

Contains the completion status for this command.

Remarks The following remarks apply:

1. The default name of the system task is SCF_SERVER. You will need to know the system task name for any subsequent deactivation or reactivation of the SCF server system task.
2. This command can be entered only once per deadstart.
3. The parameters APPLICATION_NAME, CONTROL_FACILITY_NAME, and SYSTEM_TASK_NAME can be used only in one of the following combinations:
 1. Specify none of the parameters and allow the defaults to be used.
 2. Specify all three parameters, entering values other than the default values.
 3. Specify only the parameter APPLICATION_NAME, entering a value other than the default value. In this case, the name specified for APPLICATION_NAME will also be used for the CONTROL_FACILITY_NAME and SYSTEM_TASK_NAME parameters.
4. The file \$SYSTEM.SITE_CONTROLLED.PROCEDURES.TERMINAL is a file of command procedures. Procedures on this file may contain commands for DEFINE_I_O_STATION and DEFINE_USER_I_O_STATION. Each of these commands has a CONTROL_FACILITY parameter. In order for the I/O stations to communicate with the NOS/VE SCF server, the value specified for this parameter on both commands must match the value of the ACTIVATE_SCF CONTROL_FACILITY_NAME parameter.
5. The SCF logging capability (controlled by the LOGGING parameter) is normally used only for debugging purposes. It is recommended that you enable the SCF log only when advised to do so by a Control Data customer engineer or analyst.

D.4 DEFINE_PTF Command

Purpose Defines the PTF network applications.

Format DEFINE_PTF
FAMILY_NAMES=list of name
STATUS=status variable

Parameters FAMILY_NAMES or FAMILY_NAME or FN

Defines all families that can be accessed by PTF/QTF. Up to 26 family names can be specified. This is a required parameter.

STATUS

Contains the completion status of this command.

D.5 DEFINE_SCF Command

Purpose Defines the SCF network applications.

Format DEFINE_SCF
STATUS=status variable

Parameters STATUS

Contains the completion status of this command.

D.6 DEFINE_SCFS Command

Purpose Defines the SCF server network application.

Format DEFINE_SCFS
APPLICATION_NAME=name
STATUS=status variable

Parameters APPLICATION_NAME or AN

Specifies the SCF server application name. The default name is
OSA\$STATUS_CONTROL_FAC_SERVER.

STATUS

Contains the completion status of this command.

D.7 SETUP_INSTALLATION_PROCESS Command

Purpose This command performs the following functions:

1. Creates a NOS/VE family and user name. As an example, the following command creates family NVE1 with a family administrator user name of USER1 and a password of USER1X (the system creates the password by appending the character X to the specified user name): Exception: if the value specified for the UN parameter is a full 31 characters long, X is not appended to the name.
2. Executes a CHANGE_JOB_ATTRIBUTE_DEFAULTS command to define the default family. CHANGE_JOB_ATTRIBUTE_DEFAULTS is described in the NOS/VE System Performance and Maintenance manual.
3. Automatically creates catalog \$SYSTEM.SITE_OS_MAINTENANCE and the DEADSTART_COMMANDS subcatalog and governs the file and catalog permissions for the catalogs created. You may use CHANGE_TERMINAL_ENVIRONMENT to rescind these permissions. The file and catalog permits governed by this command are:
 - a. READ access to file \$SYSTEM.MAINFRAME.CONFIGURATION.
 - b. READ, SHORTEN, APPEND, EXECUTE, and CYCLE access to catalog \$SYSTEM.SITE_OS_MAINTENANCE.DEADSTART_COMMANDS.
 - c. READ and EXECUTE access to file \$SYSTEM.SOFTWARE_MAINTENANCE.RAF\$LIBRARY.
 - d. READ and EXECUTE access to catalog \$SYSTEM.NOSVE_MAINTENANCE.DEADSTART_COMMANDS.
 - e. READ and EXECUTE access to file \$SYSTEM.MANUALS.SITE_ANALYST_EXAMPLES.
 - f. READ and EXECUTE access to file \$SYSTEM.MANUALS.EXAMPLES_FILES.SITE_ANALYST_SOURCE_LIBRARY.

Format SETUP_INSTALLATION_PROCESS
FAMILY_NAME=name
USER_NAME=user name
STATUS=status variable

Parameters FAMILY_NAME or FN

Specifies the name of the family. The name specified must be a valid SCL name and must be unique among family names on the system. For sites implementing the Permanent File Transfer (PTF/QTF) facility, the family name must also be unique among the family names for all linked systems. This is a required

parameter.

USER_NAME or UN

Specifies a user name and this name must be a valid SCL name.
This is a required parameter.

STATUS

Contains the completion status of this command.

Remarks The following remarks apply:

1. If you want to create additional users, you may do so (after you have completed this installation step) by logging into the configuration terminal and entering the Family Administration Utility. See the NOS/VE Family Administration manual for more information on this utility.
2. If you are upgrading from an earlier NOS/VE release and you intend to install the PTF/QTF product, you may have to create new family names for one or more families in order to provide unique family names. NOS/VE provides a procedure called RESTORE_FAMILY that allows you to restore a backed up family to a new family with a different name. RESTORE_FAMILY is documented in the NOS/VE Operations Manual (60463914).

D.8 SETVE Procedure

Purpose Creates a dual-state procedure file to initiate an installation or continuation deadstart of NOS/VE.

Format X.SETVE (for NOS) or n.X SETVE (for NOS/BE)
PN=procname
UN=username
P=password
C=integer (octal)
W=value
T=type
S=integer

Parameters PN

Specifies the name of the deadstart procedure file. You can specify up to four alphanumeric characters. These characters are appended to the name NVE to form a name of the following format:

NVExxxx

If this parameter is omitted, the procedure file name NVE is used. To avoid confusion between a deadstart procedure file name and the name of the NVE subsystem, it is recommended that the default name not be used as a deadstart procedure file name.

UN

Defines the NOS user name (for NOS/BE defines the permanent file ID) under which the NVE subsystem runs. Termination dumps and job dayfiles are written to the catalog associated with this user name. This parameter is required.

For first time installations of NOS/VE and NOS, user name NVE may be used (user name NVE is created automatically by the NOS installation tool SYSGEN for the purpose of running NOS dual-state).

P

Specifies the password associated with the user name defined by the UN parameter. Omission of this parameter results in the use of a password formed by appending an X to the user name specified on the UN parameter.

The P parameter has no function on NOS/BE dual-state systems, but is nevertheless required for compatibility with the NOS SETVE procedure.

C

Specifies the ordinal number portion (that is, the last two characters) of the name of the set of system core commands to be used at deadstart (refer to the DCFIL description in System Performance and Maintenance Manual for more information on system core command sets). If this parameter is omitted, the default value 01 is used.

W

Specifies whether or not the deadstart process will pause for operator entry of system core commands, for manual entry of the PCU and LCU utilities, and for operator intervention in the network initiation process. Note that if a tape or alternate disk deadstart is specified on the T parameter, the deadstart will always pause for the deadstart boot display. If this parameter is omitted, the value TRUE is used. The options are:

TRUE or T

Deadstart pauses for operator intervention.

FALSE or F

Deadstart does not pause operator intervention.

T

Specifies the type of deadstart to be performed. If this parameter is omitted, an automatic (CR) deadstart is assumed. The options are:

CR

Automatic deadstart (also called a carriage return or default deadstart); the system is deadstarted from the system disk.

DISK or D

Deadstart from disk; the system is deadstarted from an alternate disk. This option is used primarily for maintenance purposes.

TAPE or T

Deadstart from tape.

S

Specifies on which port of the 2-port multiplexor the NOS/VE system console resides. If this parameter is omitted, port 0 is assumed. If the dual-state partner system was deadstarted using the CC634B console, this option is ignored and the same console is

used for both dual-state partner and NOS/VE. The options are:

- 0 Port 0
- 1 Port 1

Remarks The following remarks apply:

1. To call the SETVE procedure from the NOS system console using DSD, you must use the following format:

X.SETVE(parameters)

To call the SETVE procedure from the NOS/BE system console using DSD, you must use the following format:

n.X SETVE(parameters)

where "n" is the control point number.

2. NOS and NOS/BE restrict the number of input characters to 50. Therefore, if you want to enter all parameters for the SETVE procedure, you may have to enter the parameters positionally, omitting the parameter names.

D.9 SET_VERSION_LEVEL Command

Purpose This command changes the default version of software to be loaded into each DI and the default version level of the three CDCNET host utilities.

Format SET_VERSION_LEVEL
OPTION = list of keyword value
VERSION = integer
STATUS = status variable

Parameters OPTION

Specifies which defaults are being set. This is a required parameter. Values that can be specified are:

UTILITIES or U

Sets the default version for the three host utilities:

ANALYZE_CDCNET_DUMP
MANAGE_CDCNET_CONFIGURATION
NETWORK_PERFORMANCE_ANALYZER

The file

\$SYSTEM.CDCNET.VERSION_INDEPENDENT.CDCNET_VERSION
is updated.

EXCEPTION_LIST or ELIST or EL

Sets the default version for software to be loaded into each DI. The file

\$SYSTEM.CDCNET.SITE_CONTROLLED.EXCEPTION_LIST
is updated.

ALL

Sets the default version for both the utilities and the exception list.

VERSION

The 4-digit hexadecimal version level. When specifying the level, you do not need to specify the radix (16). This is a required parameter.

STATUS

Contains the completion status of this command.

Remarks The following remarks apply:

1. When you update the exception list, the command searches for the string dv = vvvv(16). If this string is not in the file, the command does not update the exception list. SETVL also does not change the levels specified on any DEFINE_EXCEPTION_SYSTEM commands.
2. This command also updates the CDCNET_VERSION variable used to control the default level of the three CDCNET host utilities:
 ANALYZE_CDCNET_DUMP
 MANAGE_CDCNET_CONFIGURATION
 NETWORK_PERFORMANCE_ANALYZER
3. To use the SETVL command, you must first add it to your command list. Use the following command:

```
set_command_list a= ..  
$system.cdcnet.version_independent.command_library
```

Appendix E

Critical Display Window Messages

This appendix documents the messages that may appear in the Critical Display Window and the System Messages Line (the top line of the display) at the NOS/VE system console. These messages generally refer to a problem for which some attention should be devoted, for example, an unrecovered disk error or mainframe element error.

If a system interrupt does occur because of a failure reported to the Critical Display Window, it is very important that this information be written down to facilitate repair. Deadstarting NOS/VE after the failure will erase the contents of the Critical Display Window.

Normally, only the last two lines of the Critical Display Window are always visible; however the size of the window may be expanded as follows:

- a. Press the TAB key several times until the cursor is within the Critical Display Window.
- b. Simultaneously press the SHIFT and SUPER keys to expand the window. The SUPER key is also marked F9 at the base of the key.

The messages are grouped into the sections:

1. unrecovered disk errors
2. NOS/VE monitor detected errors
3. other errors

E.1 Unrecovered Disk Errors

All unrecovered disk failures are displayed in the Critical Display Window, regardless of the impact the failure has on the system. If the system can continue executing CP tasks, the failure data will also be put in the NOS/VE Engineering Log.

Recovered media failures and successful i/o request retries are not displayed.

If the system cannot sustain task execution, the failure data will only be available in the Critical Display Window, i.e. HPA/VE is not available to format the failure data because this process depends upon an executing NOS/VE system.

The format of the message that is displayed for unrecovered disk errors is the following:

```
CHnn Cc Uuu - <severity> <operation>, Cccc Ttt Sss  
<statement(s)>  
GSgggg, DS dddd dddd dddd dddd dddd dddd dddd dddd dddd dddd  
          dddd dddd dddd dddd dddd dddd dddd dddd dddd dddd
```

The actual content of the message depends on the disk controller which received the error. The following sections describe the message content for each type of disk controller.

E.1.1 7155-1x Controller

For a 7155-1x controller, the message

```
CHnn Cc Uuu - <severity> <operation>, Cccc Ttt Sss  
<statement(s)>  
GSgggg, DS dddd dddd dddd dddd dddd dddd dddd dddd dddd dddd  
          dddd dddd dddd dddd dddd dddd dddd dddd dddd dddd
```

has the following definitions.

CHnn	Channel connected to the controller (in decimal).
Cc	The 7155-1x controller address (this is always 0).
Uuu	The decimal unit number of the 844-4x disk (0 through 7). or the 885-1x disk (32 through 47)
severity	The string UNRECOVERED for the message documenting the last unsuccessful attempt at retrying a failing disk request; otherwise the <severity> is a null string.

operation Can be either READ, WRITE, or READ FLAW MAP.

Cccc Decimal cylinder address of the failing operation.

Ttt Decimal track address of the failing operation.

Sss Decimal sector address of the failing operation. This is a logical address which because of sector relocation within the track may be zero, one, or two sectors less than the physical address. The physical address is the one used for off-line sector relocation.

The Cccc, Ttt, and Sss values can be used for flawing or relocation.

statement One of the failure symptoms documented in the CM4101 Engineering Log statistic. This statistic is documented in the NOS/VE System Performance and Maintenance Manual (60463915).

gggg The General Status from a 7155-1x controller. The status is displayed in an octal representation.

dddd Twenty words of detailed status from a 7155-1x controller. The status is displayed in an octal representation. Words 1 through 10 of detailed status are displayed on the first line and words 11 through 20 are displayed on the second line.

An octal representation of the cylinder, track, and sector values is in the following words of the detailed status.

844-4x	1110 9 8 7 6 5 4 3 2 1 0	
word 4		U U U U U U (U= unit number)
5	C C C C C C C C T T T	
6	T T S S S S S	C (Cylinder,Track,Sector)

885-1x	1110 9 8 7 6 5 4 3 2 1 0	
word 4		U U U U U U (U= unit number)
5	C C C C C C	
6	C C C C T T T T T T T T	(Cylinder,Track,Sector)
7	S S S S S S S S	

E.1.2 10395-11 and FA7B4-D Controllers

For the 10395-11 and FA7B4-D controllers, the message

```
CHnn Cc Uuu - <severity> <operation>, Cccc Ttt Sss  
<statement(s)>  
GSgggg, DS ddd ddd ddd ddd ddd ddd ddd ddd ddd ddd  
          ddd ddd ddd ddd ddd ddd ddd ddd ddd ddd
```

has the following definitions.

CHnn	Channel connected to the controller (in decimal).
Cc	The 10395-11 or FA7B4-D controller address (0..7)
Uuu	The unit number of an 834-12 or an 836-xxx disk unit (0..3).
severity	The string UNRECOVERED for the message documenting the last unsuccessful attempt at retrying a failing disk request; otherwise the <severity> is a null string.
operation	Can be either READ or WRITE.
Cccc	Decimal cylinder address of the failing operation.
Ttt	Decimal track address of the failing operation.
Sss	Decimal sector address of the failing operation. The address provided for the 834-12 is the address of a 512 byte sector doublet, which is the unit of re-location for this device. The Cccc, Ttt, and Sss values are a physical media address which may be used for media flawing or relocation.
statement	Can be one of the following informational statements which indicate actions begun or completed. These actions may affect the real-time response of interactive jobs.

```
RELOADING CONTROLWARE  
CONTROLWARE RELOADED  
POWERING UP SPINDLE  
SPINDLE POWERED UP
```

The 10395-11 and FA7B4-D controllers have diagnostic self-tests which may take as long as 2.5 minutes to execute. Messages report initiation of these diagnostics and the normal or abnormal completion status of the diagnostic.

EXECUTING LEVEL II DIAGNOSTICS
LEVEL II DIAGNOSTICS PASSED

The following failure descriptions indicate that the failure was isolated to a particular subsystem element by executing subsystem diagnostics. The "code" is the 1..3 hex digit result of the failing diagnostic. Note that this is reported so that if the system should subsequently fail, the information would be available. These messages DO NOT MEAN THAT THE SYSTEM HAS FAILED.

ADAPTER FAILURE, code
CONTROL MODULE FAILURE, code
DRIVE FAILURE, code

The following failure descriptions indicate that the failure was isolated to a particular subsystem element and that the failure was not tolerated. The system disabled further use of the element. A maintenance action is required.

ADAPTER DOWNED
CONTROL MODULE DOWNED
DRIVE DOWNED

The following failure description indicates that the failure is presumed to have been associated with a particular media address; however, the failure could have been an intermittent problem in the storage device or elsewhere. The Level II diagnostic 72, subtest 5 successfully wrote and verified data on all tracks of the maintenance cylinder. Because of this, the problem appears to be isolated to a bad sector. The storage device is not downed as a result of a media failure.

MEDIA FAILURE

If the message is not one of the preceding messages, then the <statement> is one of the failure symptoms documented in the CM4102 Engineering Log statistic (10395-11) or the CM4103 statistic (FA7B4-D). These statistics are documented in the NOS/VE System Performance and Maintenance Manual (60463915).

The failure status may not be displayed for some failure modes, e.g. ADAPTER DOWNED.

gggg The poll status from a 7255-1 adapter. The status is displayed in a hexadecimal representation. The status may not be displayed for some failure modes, e.g. ADAPTER DOWNED.

ddd Twenty words of detailed status from a 7255-1. The status is displayed in a hexadecimal representation. Words 1 through 10 of detailed status are displayed on the first line and words 11 through 20 are displayed on the second line. The status may not be displayed for some failure modes, e.g. ADAPTER DOWNED.

A hexadecimal representation of the cylinder, track, and sector values is in the following words of the detailed status.

834-12 and 836-12:

word 12	C C C	(C= cylinder)
13	T T S	(T= track)
14	S	(S=sector)

NOTE

The address reported in detailed status for the 834-12 failure is that of a 256 byte sector. However, sector re-location is only done on 512 byte sector doublets. Therefore, the reported address must be divided by two to perform sector re-location.

E.1.3 7165-2x Controller

For a 7165-2x controller, the message

CHnn Cc Uuu - <severity> <operation>, Cccc Ttt Sss
<statement>
GSgggg, DS ddd ddd ddd ddd ddd ddd ddd ddd ddd ddd
ddd ddd ddd ddd ddd ddd ddd ddd ddd ddd

has the following definitions.

CHnn Channel connected to the controller (in decimal).

Cc The 7165-2x Storage Director address (0 or 1).

Uuu The decimal unit number which is formed from the 895-1 (DDC) address and the unit address as follows:

$$uu = (\text{DDC address}) * 16 + \text{disk unit address}$$

If uu is less than 16, the DDC address is zero; otherwise the DDC address is one.

severity The string UNRECOVERED for the message documenting the last unsuccessful attempt at retrying a failing disk request; otherwise the <severity> is a null string.

operation Can be either READ or WRITE, or SOFT SECTORING.

Cccc Decimal cylinder address of the failing operation.

Ttt Decimal track address of the failing operation.

Sss Decimal sector address of the failing operation. The address provided for the 895-2 is the address of a 4128 byte sector.

statement Can be categorized as either informational, failure description, or other.

These categories are discussed next.

INFORMATIONAL

A statement that indicates actions begun or completed. These actions may affect the real-time response of interactive jobs.

SOFT SECTORING UNIT
UNIT SOFT SECTORED

FAILURE DESCRIPTIONS

These indicate that the failure was isolated to a particular subsystem element and that the failure was not tolerated. The system disabled further use of the element. A maintenance action is required.

CCC DOWNED
SD/HSC DOWNED
HDA DOWNED

OTHER

If the statement is not one of the preceding messages, then one of the failure symptoms documented in the CM4104 Engineering Log statistic is displayed. This statistic is described in the NOS/VE System Performance and Maintenance Manual (60463915).

gggg The general status from the CCC. The status is displayed in hexadecimal representation. This status is not displayed for all failures. The status may not be displayed for some failure modes, e.g. SD/HSC DOWNED.

ddd Twenty (12 bit) words of detailed status from the CCC. The status is displayed in hexadecimal representation. Words 1 through 10 of detailed status are displayed on the first line and words 11 through 20 are displayed on the second line. The status may not be displayed for some failure modes, e.g. SD/HSC DOWNED.

```

895-2          1110 9 8 7 6 5 4 3 2 1 0
      word  4      U U U U C C C C C C C C (U = unit )
              5          C C   T T T T      (C=cylinder)
              9              S S S S      (T = track )
             10      S S S S      (S = sector)

```

E.2 NOS/VE Monitor Detected Errors

The NOS/VE monitor detects such errors as mainframe errors, environment conditions, and monitor conditions. The messages listed in this section always appear at the operator console System Message Line (the top line of the display). The description of each message mentions any other message and where they may appear (the console Critical Display Window and the Engineering Log).

E.2.1 (004) UNCORRECTED IOU ERROR

An uncorrected error in a specific nonconcurrent peripheral processor was detected by the IOU during operations of the system.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to an UNCORRECTED IOU ERROR'

The message in the Critical Display Window is: 'System stepped due to an UNCORRECTED IOU ERROR'

E.2.2 (009) UNCORRECTED IOU ERROR

An uncorrected error in a specific concurrent peripheral processor was detected by the IOU during operations of the system.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to an UNCORRECTED IOU ERROR'

The message in the Critical Display Window is: 'System stepped due to an UNCORRECTED IOU ERROR'

E.2.3 (004) C170 STATE OS IOU ERROR

An uncorrected error in a specific NOS or NOS/BE nonconcurrent peripheral processor was detected by the IOU during operations of the system.

NOS/VE does not STEP due to this condition.

E.2.4 (009) C170 STATE OS IOU ERROR

An uncorrected error in a specific NOS or NOS/BE concurrent peripheral processor was detected by the IOU during operations of the system.

NOS/VE does not STEP due to this condition.

E.2.5 (006) FATAL IOU ERROR

A fatal IOU error was detected during operations of the system. The error occurred in a nonconcurrent peripheral processor.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a FATAL IOU ERROR'

The message in the Critical Display Window is: 'System stepped due to a FATAL IOU ERROR'

E.2.6 (008) FATAL IOU ERROR

A fatal IOU error was detected during operations of the system. The error occurred in a concurrent peripheral processor.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a FATAL IOU ERROR'

The message in the Critical Display Window is: 'System stepped due to a FATAL IOU ERROR'

E.2.7 (105) FATAL CM ERROR

A multiple odd bit central memory error was detected during operations of the system.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a MULTIPLE ODD BIT CENTRAL MEMORY ERROR'

The message in the Critical Display Window is: 'System stepped due to a MULTIPLE ODD BIT CENTRAL MEMORY ERROR'

E.2.8 (20E) FATAL CPU ERROR

A fatal detected uncorrected error (DUE) was detected in a NOS/VE job during operations of the system.

The system goes into system STEP mode and it cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a FATAL JOB DUE'

The message in the Critical Display Window is: 'System stepped due to an UNCORRECTED PROCESSOR ERROR'

E.2.9 (20F) FATAL C170 STATE UNCORRECTED CPU ERROR

A fatal detected uncorrected error (DUE) was detected in a NOS or NOS/BE job during operations of the system. The system is no longer running NOS or NOS/BE; however, NOS/VE is still operating.

The message in the Critical Display Window is: '170 OS TERMINATED due to FATAL DUE'

E.2.10 (210) FATAL C170 STATE EXIT MODE HALT

NOS or NOS/BE has moded out and are no longer running; however, NOS/VE is still operating.

The message in the Critical Display Window is: '170 OS TERMINATED due to MODE OUT error'

E.2.11 (211) FATAL MONITOR DUE

A fatal detected uncorrected error (DUE) was detected in NOS/VE monitor mode during operations of the system. The number of tasks which were aborted exceeded a threshold and the system could not continue to execute with confidence.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a FATAL MONITOR DUE'

The message in the Critical Display Window is: 'System stepped due to an UNCORRECTED PROCESSOR ERROR'

E.2.12 (212) error

A fatal software error in NOS/VE monitor was detected during operations of the system, where 'error' is a string describing the error.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a MONITOR ERROR STOP'

The message in the Critical Display Window is: 'System stepped due to a MONITOR ERROR STOP'

E.2.13 (213) FATAL MTR MCR

A fatal monitor condition was detected in NOS/VE monitor mode during operations of the system.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a FATAL MONITOR MCR'

The message in the Critical Display Window is: 'System stepped due to a FATAL MONITOR MCR'

E.2.14 (216) error

A software error in a NOS/VE job during operations of the system caused the system to step, where 'error' is a string describing the error.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to SOFTWARE ERROR BELOW CPU HALT_RING'

The message in the Critical Display Window is: 'System stepped due to a SOFTWARE ERROR BELOW CPU HALT_RING'

E.2.15 HR - (217) job mode MCR fault

A fatal condition was detected in NOS/VE job mode during operations of the system.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on

MM/DD/YY. Reason: System stepped due to SOFTWARE ERROR BELOW CPU HALT_RING'

The message in the Critical Display Window is: 'System stepped due to a SOFTWARE ERROR BELOW CPU HALT_RING'

E.2.16 (701) ENVIRONMENT WARNING

Changing environment conditions or power warnings were detected during operations of the system.

The system can be RESUMED from this condition when the condition clears.

The message in the Engineering Log is: 'SYSTEM IDLE at HH:MM:SS.SSS on MM/DD/YY. Reason: System idled due to an ENVIRONMENT WARNING'

The message in the Critical Display Window is: 'System idled due to an ENVIRONMENT WARNING'

E.2.17 (702) POWER WARNING

A long warning was detected during operations of the system.

The system can be RESUMED from this condition when the condition clears.

The message in the Engineering Log is: 'SYSTEM IDLE at HH:MM:SS.SSS on MM/DD/YY. Reason: System idled due to a LONG WARNING'

The message in the Critical Display Window is: 'System idled due to a LONG WARNING'

E.2.18 (703) POWER WARNING

A short warning was detected during operations of the system.

The system can be UNSTEPPED from this condition when the condition clears.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped due to a SHORT WARNING'

The message in the Critical Display Window is: 'System stepped due to a SHORT WARNING'

E.2.19 (704) POWER/ENVIRONMENT NORMAL

A previous environment warning condition which caused the system to IDLE or STEP has cleared. The system can resume normal operations, either automatically or by operator command.

E.2.20 (705) POWER/ENVIRONMENT NORMAL

A previous long warning condition which caused the system to IDLE or STEP has cleared. The system can resume normal operations, either automatically or by operator command.

E.2.21 (706) POWER/ENVIRONMENT NORMAL

A previous short warning condition which caused the system to IDLE or STEP has cleared. The system can resume normal operations, either automatically or by operator command.

E.2.22 TERMINATE NVE COMMAND

A TERMINATE_SYSTEM command was invoked.

The system cannot be RESUMED from this state.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped via TERMINATE NVE COMMAND'

The message in the Critical Display Window is: 'System idled/stepped via TERMINATE NVE COMMAND'

E.2.23 RECOVERY SYSTEM NEEDED

A NOS/VE system recovery was needed before the system could continue any further. NOS/VE recovery terminated abnormally and put out this message before beginning recovery again.

The system cannot be UNSTEPPED from this condition. However, this condition is temporary, as the system will begin recovery of NOS/VE again.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped because a RECOVERY SYSTEM IS NEEDED'

The message in the Critical Display Window is: 'System stepped because a RECOVERY SYSTEM IS NEEDED'

E.2.24 RECOVERY SYSTEM DONE

A NOS/VE recovery was completed. NOS/VE is now able to terminate quietly or perform a continuation deadstart.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped because RECOVERY SYSTEM IS DONE'

The message in the Critical Display Window is: 'System stepped because RECOVERY SYSTEM IS DONE'

E.2.25 180 terminated by 170 via SCB request

This message only applies to dual-state systems. The command DROPVE was entered from NOS or NOS/BE.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped via DROPVE COMMAND'

The message in the Critical Display Window is: 'System stepped via DROPVE COMMAND'

E.2.26 Dual-state CPU was halted

The processor used to run dual state in a multiple CPU system was halted.

The system cannot be UNSTEPPED from this condition.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on MM/DD/YY. Reason: System stepped because the 170 CPU WAS TURNED OFF'

The message in the Critical Display Window is: 'System stepped because the 170 CPU WAS TURNED OFF'

E.2.27 error message

A software error was detected as occurring below the CPU halt-ring, where 'error' is the message describing the error.

The system can be UNSTEPPED from this state.

The message in the Engineering Log is: 'SYSTEM STEP at HH:MM:SS.SSS on

NOS/VE 1.2.1 L670 SRB (11/19/86)
Critical Display Window Messages

E-16
NOS/VE Monitor Detected Errors

MM/DD/YY. Reason: System stepped due to SOFTWARE ERROR BELOW CPU HALT_RING'

The message in the Critical Display Window is: 'System stepped due to
SOFTWARE ERROR BELOW CPU HALT_RING'

E.3 Other Messages

Messages which don't fall into the unrecovered disk errors or monitor detected errors categories are documented in this section.

E.3.1 TASK IS HUNG. HALT OPTION SELECTED.

This message appears when the HALT_ON_HUNG_TASK system attribute is 1. Refer to the NOS/VE System Performance and Maintenance Manual (60463915) for more information.

E.3.2 n VOLUMES LOW, n VOLUMES OUT date time

Summarizes the number of disk volumes which are either low on space or out of space. The 'date' and 'time' values are when the message was displayed.

The VEDISPLAY MASS_STORAGE display can be used to display detailed information about the status of all disk volumes.

E.3.3 n VOLUMES OUT date time

Summarizes the number of disk volumes which are out of space. The 'date' and 'time' values are when the message was displayed.

The VEDISPLAY MASS_STORAGE display can be used to display detailed information about the status of all disk volumes.

E.3.4 PURGED CATALOG path-name DELETED

This message indicates a catalog was deleted when recovering permanent files with job recovery enabled. This condition can only affect the ability of a job to recover and does not indicate a problem in permanent file recovery.

Appendix F

Peripheral Reconfiguration and Online Repair

The purpose of this appendix is to outline how the site would deal with common peripheral failure modes in NOS/VE R1.2.1.

To realize the greatest benefit from the reconfiguration capabilities offered by NOS/VE R1.2.1, the site must restrict its users' catalog residence to a small number of volumes (including the system device, which must be a catalog device) using the LCU's CHANGE_MS_CLASS subcommand. Only devices not containing catalogs can be removed from a configuration without requiring an installation deadstart. Having only one catalog device (that is, the system device) may cause a performance degradation because catalog references may conflict with system file access. However, having more than two catalog devices increases the probability of losing catalogs and, therefore, of losing the entire file base.

A catalog device may contain permanent or temporary files as well as catalogs. If possible, keep permanent and temporary files off catalog devices to ensure enough space for file base recovery and catalog growth. Again the LCU subcommand CHANGE_MS_CLASS may be used to control permanent and temporary file residence.

If possible, configure dual-access 7155 and 7165 mass storage subsystems for greater availability. One access can be repaired online concurrent with customer use of the remaining access. Dual access is particularly important to the system disk and to catalog devices both for availability and for performance.

After installation of NOS/VE release materials or after a BCU upgrade, backup the \$SYSTEM catalog hierarchy to tape for use during repair solution 2 which is described later in this appendix.

F.1 Capabilities

NOS/VE will allow a channel or controller to be put in either the DOWN or the OFF state during system execution if, and only if, access to an active storage device is not denied as a result of the action.

NOS/VE can be operated indefinitely with one or more missing volumes, and the volumes may later be reinstated on a continuation deadstart without losing files for which no backup exists. File cycles residing on the missing volume are marked as being in a "media missing" state until one of the following things happen:

1. The file cycle is deleted by a user.

2. The file is deleted by the site administrator or a user with the DELETE_LOST_CYCLES parameter of the CHANGE_CATALOG_CONTENTS command. This command is made available on \$SYSTEM.OSF\$SITE_COMMAND_LIBRARY.
3. The file cycle is restored from a backup tape using the RESTORE_EXCLUDED_FILE_CYCLES subcommand of RESTORE_PERMANENT_FILE.
4. The failure is repaired and the missing volume is reinstated. At each subsequent continuation deadstart, the operator will have an opportunity to reinstate the missing volume(s) to the set.

Omitting a device that contains no catalogs from the active configuration is a reversible operation (that is, no files are permanently deleted). Therefore you may experiment by removing a volume during a NOS/VE continuation deadstart to determine if, in doing so, you can recover files on the remaining volumes. NOS/VE reports to the system log the full path name of each file which is no longer accessible and which has been placed in a "media missing" state.

Loss of the system disk or any catalog disk requires an installation deadstart and subsequent restoration of all permanent files.

A continuation deadstart is required to logically remove disk volumes from the configuration and to reinstate them.

Jobs which were active prior to a system interrupt or a TERMINATE_SYSTEM operator command are not likely to be recovered if any volumes which were in the set during system execution are omitted from the configuration. For an active job to be recovered, all the files which it had attached and all its temporary files must be accessible during the continuation deadstart that immediately follows a system interruption. Jobs whose swap image resides on an omitted volume also cannot be recovered.

MALET/VE provides online diagnosis of a failure concurrent with customer use of the system.

F.2 Guidelines

The following figures can be used to determine the recommended course of action in response to a hardware or software failure in a mass storage or magnetic tape subsystem.

NOS/VE reports all unrecoverable mass storage failures to the "Critical Window" of the system console. Use the shifted F9 (SUPER) key to expand the size of this window to allow you to see more information. Many of the following suggested repair solutions require a deadstart with pause for operator input. For standalone NOS/VE, a deadstart with operator pause is selected by entering the P=YES option on the CYBER Initialization Package (CIP) DEADSTART PANEL PARAMETERS display. Refer to chapter ?? or to the CYBER Initialization Package (CIP) Manual (60457180) for more information on CIP displays.

For dual-state NOS/VE, deadstart with operator pause is initiated by executing an appropriate NVExxxx procedure. Refer to the SETVE procedure description in chapter ?? (for NOS dual-state) or chapter ?? (for NOS/BE dual-state) for information on how to create an appropriate NVExxxx procedure.

F.3 Repair Solution 1

This solution deals with mass storage controller and channel cable failures in a dual access 7155 or 7165 subsystem when the mass storage volumes have not been damaged by the failure. If you have a 10395-11 (836), FA7B4-D (836), 7155-1 or 7155-1x controller connected to more than one channel and the channel or the 7255-1 channel adapter you were using failed, you may logically reconfigure NOS/VE to use one of the remaining channels; refer to Repair, Solution 3.

If the failure to a channel or controller does not cause a system interrupt, then it is possible to manually DOWN the failing element and perform online diagnosis, repair, and reinstatement. If the failure caused a system interruption, then a continuation deadstart must be performed. During the deadstart, the state of the failing element may be changed to DOWN and normal system execution can continue, but perhaps with degraded performance. Subsequent to the interrupt, the failing element may be repaired and turned back ON without requiring a deadstart.

F.3.1 Repair of 7155 Channel/Controller

In the following discussion, assume volumes VSN007 and VSN008 are the only mass storage volumes connected to the failing 7155 channel or controller.

1. Use CHANGE_ELEMENT_STATE LCU subcommand to change the state of the failing channel or controller to DOWN.

```
change_element_state green_7155_1 state=down
```

2. Backup files located on the volumes connected to the failing channel or controller. This is an insurance policy should repair damage the media.

```
request_magnetic_tape file=$local.dump ..  
external_vsn=(('reel1'), ('reel2')) ..  
type=mt9$6250 ..  
ring=true  
backup_permanent_files backup_file=$local.dump  
include_volumes recorded_vsns=(VSN007,VSN008)  
backup_all_files  
quit
```

3. Contact a CE to diagnose and repair the problem while you are using the system.
4. To reinstate the repaired channel or controller, use the `CHANGE_ELEMENT_STATE` LCU subcommand to change the state of the element back to ON.

```
change_element_state green_7155_1 state=on
```

F.3.2 Repair of 7165 Channel/Controller

In the following discussion, assume volumes VSN007 and VSN008 are the only mass storage volumes connected to the failing 7165 channel or controller.

1. If the failure is in the 7165 channel, use `CHANGE_ELEMENT_STATE` LCU subcommand to change the state of the channel to DOWN. If the failure is in a Storage Director (SD), DOWN the channel connected to the SD to prevent NOS/VE from using the channel and the SD.

```
change_element_state ch5 state=down
```

2. Backup files located on the volumes connected to the failing channel or controller. This is an insurance policy should repair damage the media.

```
request_magnetic_tape file=$local.dump ..  
  external_vsn= (('reel1'), ('reel2')) ..  
  type=mt9$6250 ..  
  ring=true  
backup_permanent_files backup_file=$local.dump  
  include_volumes recorded_vsns=(VSN007,VSN008)  
  backup_all_files  
quit
```

3. Contact a CE to diagnose and repair the problem while you are using the system.
4. To reinstate the repaired channel or controller, use the `CHANGE_ELEMENT_STATE` LCU subcommand to change the state of the channel back to ON.

```
change_element_state ch5 state=on
```


F.4 Repair Solution 2

This repair solution is used when the system is no longer operational as a result of one of the following conditions:

1. A mass storage channel cable failure or a controller or drive logic failure.
2. An HDA failure (head crash, bad media, etc.).
3. A software failure which prevents customer data on a volume from being used or recovered.

This solution offers the following benefits which would not be realized if you performed an installation deadstart (Repair Solution 4):

1. You only need to restore the number of files that gets you quickly back into production, rather than having to re-install NOS/VE and restore all files and catalogs.
2. You can recover files which had been modified since the last backup but prior to the peripheral failure. These files would be lost, necessitating rework, if you use repair solution 4.
3. For other volumes, you can retain the media defect list maintained by NOS/VE on each device. Media defects automatically flawed by NOS/VE are not retained when you do an INITIALIZE_MS_VOLUME LCU subcommand. Therefore you may re-encounter the media defects you found earlier.
4. Volumes do not require initialization. On an 895-2 volume, this avoids the necessity of rewriting sector addresses.

This solution may have the following disadvantages compared to an installation deadstart (Repair Solution 4):

1. If a NOS/VE system interrupt should occur, the system will recover files and jobs which were in central memory prior to the failure, unless central memory has been lost or compromised. Modified file data which has not yet been written to disk can be recovered. This recovery is only possible if the mass storage volume(s) to which a file is assigned is accessible at the time the continuation deadstart is performed. However, if a volume is omitted from the configuration on a continuation deadstart, modified data cannot be written to the missing volume and the data is lost. When the volume is reinstated the files which were attached for write at the time of the failure may have inconsistent data. At NOS/VE R1.2.1 there is no notification that modified data for a file has been lost due to the loss of central memory or to the loss of a disk volume.
2. Your file base may become inconsistent if you restore files whose modification date does not match the catalog.

For example, assume you have two files called SOURCE and OBJECT_LIBRARY. Ordinarily SOURCE would accurately represent the contents of OBJECT_LIBRARY. However, if SOURCE is on a volume which is not currently accessible and you restore an older version of SOURCE using this repair solution, the two files will no longer match.

In the following repair solution, assume that the mass storage element named BLUE_DISK_42, whose media has a recorded VSN of VSN007, has failed and must be removed from the active configuration:

1. Deadstart NOS/VE, requesting pause for operator input.
2. Enter the GO (rather than AUTO) system core command to terminate system core command processing.
3. Use the PCU to change the state of BLUE_DISK_42 to DOWN. This prevents the device from being accessed during file base recovery.

To the question:

Do you want to change the mainframe's physical configuration?
Enter yes or no:

Answer YES or Y. Then enter:

```
edit_physical_configuration
change_element_definition element=BLUE_DISK_42 state=down
quit
install_physical_configuration
quit
```

4. The next operator pause is for execution of the Logical Configuration Utility. You would use the LCU to change the state of a failing channel to DOWN using the CHANGE_ELEMENT_STATE subcommand.

To the question:

Do you want to change the mainframe's logical configuration?
Enter yes or no:

Answer YES or Y if you need to down a channel; otherwise NO or N. If you answered yes, then change the channel's state as follows:

```
change_element_state ch4 state=down
quit
```

5. The next operator pause is for the execution of the Physical Configuration Utility after the completion of file base recovery.

To the question:

Do you want to change the mainframe's physical configuration?
Enter yes or no:

Answer YES if you need to turn ON standby mass storage equipment which is defined in your physical configuration but OFF; otherwise answer NO. As shown above, change the state of any standby elements using `change_element_definition`.

6. The next operator pause is for the execution of the Logical Configuration Utility. You may at this time initialize standby mass storage volumes and add them to the set.

If you are adding a standby volume to the set, you must choose a previously unused recorded VSN for each volume.

To the question:

Do you want to change the mainframe's logical configuration?
Enter yes or no:

Answer YES or Y, if you have standby volumes to add to the configuration; otherwise answer NO or N. If you answered YES, then enter the following two subcommands for each volume to be added:

```
initialize_ms_volume element=<name> recorded_vsn=<name>
add_volume_to_set member_vsn=<name>
quit
```

7. Because you have omitted a volume from the set, the following question will appear:

Do you want to delete the missing volume(s) from the set?

If you have no intention of recovering files on the inaccessible volume(s) after repair has completed, you may answer YES or Y to this question. You do not have to make the decision now; you will have an opportunity at each future continuation deadstart to make this decision. If you think you may want to recover files on the missing volume, answer NO or N.

8. Deadstart completes.

9. The system is not yet ready for production. Because you omitted one or more non-catalog volumes, the catalogs can be interrogated to provide information to determine the magnitude and the consequences of restoring files at this time. Submit the following job specifying the date and time of the last backup (full or partial). The output file will contain a list of all files which have been modified since the last backup and a "media missing" message for each file cycle affected by the failure. This output will tell you the magnitude of your users' rework effort should you decide not to reinstate the volume(s).

```
job job_name=display job_class=system
task ring=3
  backup_permanent_files backup_file=$null
  include_cycles modified_after <date-time>
  backup_all_files
  quit
  copy_file $output $system.modified_files_listing
  change_file_attributes $system.modified_files_listing ..
  ring_attributes=(13,13,13)
taskend
jobend
```

Submit the following job to obtain a list of all files affected by the failure. A "media missing" message will be provided for each file cycle whose data is on the lost volume(s). This output will tell you the magnitude of your cycle restoration effort, should you decide not to reinstate the volume.

```
job job_name=display job_class=system
task ring=3
  backup_permanent_files backup_file=$null
  backup_all_files
  quit
  copy_file $output $system.lost_files_listing
  change_file_attributes $system.lost_files_listing ..
  ring_attributes=(13,13,13)
taskend
jobend
```

Based on the information you obtained from running the preceding jobs, you now must decide whether or not to restore files at this time.

If you have not yet deleted the missing volume(s) and you still hope to recover the missing files, the only files you should consider restoring are those files required to sustain production. In this category are the files in \$SYSTEM which you may have backed up after your installation or upgrade of NOS/VE.

You have the following choices:

1. Restore all file cycles which had not been modified since the last backup.

Choosing this option allows you to ultimately recover those files which have been modified since the last backup and to provide access now to all the other missing files.

To accomplish this use the \$SYSTEM.RESTORE procedure to restore files from all the pertinent partial backup tapes and then the full backup tapes. This procedure uses RESTORE_EXCLUDED_FILE_CYCLES but does not restore cycles whose last modification date disagrees with the backup file.

```
attach file $system.restore
restore last backup=false ..
backup vsn count=6 ..
type=mt9$6250
```

2. Restore files on a demand basis.

All files which reside on the missing volume(s) are placed in a "media missing" state which prevents their attachment until the volume(s) are repaired and reinstated.

Choosing this option may take less time initially but more time later, depending upon how many files need to be restored. It all depends upon the probability that a user will reference the missing files and will want the files restored rather than waiting for the repair.

If a file to be restored was modified since the last partial backup, you will need to use the SET_RESTORE_OPTIONS subcommand of the RESTORE_PERMANENT_FILE utility to restore a file which has a differing modification date. See below how to restore a cycle which has a different modification date than the catalog.

A user can determine how many of his or her cycles are inaccessible by doing a DISPLAY_CATALOG of their master catalog or a subcatalog:

```
display_catalog $user display_option=contents ..
depth=all
```

A "MEDIA_MISSING" message will be returned for each lost cycle.

3. Reload all missing files.

Choosing this option means that you cannot recover any of the files which were modified since the last backup. This is the logical choice if the volume or volumes have been damaged and cannot be recovered anyway. This might also be a good choice if you want to get back to a well-defined point so that all the files are consistent with respect to each other.

Choosing this option and then reinstating the missing volume without initializing it means that a message will be recorded in the system log for each file that is no longer described by a catalog. These messages will be displayed only on the deadstart which reinstates the volume. Initializing the volume will inhibit the messages about lost files but will also cause the loss of software flaws maintained on the volume.

The catalogs may describe files which have been modified since the last backup but before the failure. Therefore you may have to restore a cycle which has a different modification date than the catalog. To restore all files whose last modification date differs from that defined in the catalog, use the system provided procedure:

`$$SYSTEM.RESTORE_LOST_CYCLES.`

```
attach_file $system.restore_lost_cycles
restore_lost_cycles backup_vsn_count=6 ..
type=mt9$6250
```

Each cycle restored with a modification date other than the one described in the catalog will be marked by NOS/VE, such that an abnormal status will be displayed to a user who should attach this cycle. This alerts the user to the fact that the file data may be other than what the user expects. The user may again attach the file after using the `CHANGE_CATALOG_ENTRY SCL` command to delete the `RESPF_MODIFICATION_MISMATCH` damage condition.

If you do not want your users to know that a restore of a cycle with a different modification date has occurred, then use the `CHANGE_CATALOG_CONTENTS` command provided on `$$SYSTEM.OSF$COMMAND_LIBRARY` to delete the damage condition from all cycles in each affected user's master catalog hierarchy:

```
$system.osf$site_command_library.change_catalog_contents ..
catalog=user-name ..
delete_damage_conditions=respf_modification_mismatch
```

Delete all cycles which had no backup and therefore were not restored. At the system console, repeat the following sequence of commands, once for each user affected:

```
$system.osf$site_command_library.change_catalog_contents ..
catalog=:family-name.user-name ..
delete_lost_cycles=true
```

10. Request a CE to diagnose and repair the problem while you are using the system.
11. To reinstate the missing volume(s) you must perform a continuation deadstart. However, you may schedule this at any convenient time. Do the following at the system console:
 - a. Enter a `TERMINATE_SYSTEM` operator command.
 - b. Perform a NOS/VE continuation deadstart requesting deadstart pause for operator input.
 - c. If you want to recover the files on the volume to be reinstated, enter the system core command `VALIDATE_SYSTEM_SET` and continue with step 10c; otherwise, go to step 10d.

Enter the `GO` (rather than `AUTO`) system core command to terminate core command processing.

Using the PCU, change the state of each storage device back to ON.
When the following prompt appears:

Do you want to change the mainframe's physical configuration?
Enter yes or no:

Answer YES or Y. Then enter:

```
edit_physical_configuration
change_element_definition e=<name> s=on
quit
install_physical_configuration
quit
```

When the following prompt appears:

Do you want to change the mainframe's logical configuration?
Enter yes or no:

Answer NO or N.

Then answer NO or GO (whichever is appropriate) to subsequent system prompts and complete the deadstart.

- d. If you want to initialize the volume (rather than recover its files) and then reinstate the volume, enter AUTO (rather than GO) to terminate system core command processing.

When the following prompt appears:

Do you want to delete the missing volume(s) from the set?

Answer YES or Y.

Complete the continuation deadstart. Then terminate the system and perform a continuation deadstart requesting deadstart pause for operator input.

Enter GO (rather than AUTO) to terminate system core command processing.

Using the PCU, change to ON the state of each storage device to be reinstated.

When the following prompt appears:

Do you want to change the mainframe's physical configuration?
Enter yes or no:

Answer YES or Y. Then enter:

```
edit_physical_configuration
  change_element_definition e=<name> s=on
quit
install_physical_configuration
quit
```

Using the LCU, initialize each volume to be reinstated and add each to the system set.

When the following prompt appears:

```
Do you want to change the mainframe's physical configuration?
Enter yes or no:
```

Answer YES or Y. Then enter the following pair of LCU subcommands for each volume to be added:

```
initialize_ms_volume element=<name> ..
  recorded_vsn=<name>
add_volume_to_set member_vsn=<name>
quit
```

Answer NO or GO (whichever is appropriate) to subsequent system prompts and complete the deadstart.

F.5 Repair Solution 3

This solution deals with recabling a mass storage device to a different controller or a controller to a different channel.

1. If physical recabling is required, request a CE to recable the affected peripheral elements. If you have a 10395-11 (836), FA7B4-D (836), 7155-1 or 7155-1x controller connected to more than one channel and the channel or the 7255-1 channel adapter you were using failed, you may logically reconfigure NOS/VE to use one of the remaining channels; refer to step 4, below.
2. Perform a NOS/VE continuation deadstart requesting deadstart pause for operator input.

Note that if you are operating a standalone NOS/VE system and the CIP device must be recabled or an alternate CIP device is to be used, you must change the deadstart panel setting to describe the new location of the CIP device. Refer to the CIP User's Handbook to determine what information must be specified. If the CIP device is also the NOS/VE system disk, you will also need to describe the reconfiguration of the system disk to NOS/VE. For more information on that subject, refer to the section called Reconfiguration of NOS/VE Deadstart Peripherals later in this appendix.

3. Enter GO (rather than AUTO) to terminate system core command processing.
4. Enter GO to redefine the IOU or peripheral connections of the affected storage device(s). For example, the following command sequence describes the recabling of the system disk to the BLUE_CONTROLLER at unit number 35, as well as the downing of the failing controller to which the system disk had previously been connected. To the question:

Do you want to change the mainframe's physical configuration?
Enter yes or no:

Answer YES or Y. Then enter:

```
edit_physical_configuration
  change_element_definition element=red_controller state=down
  change_element_definition element=blue_885_35 ..
    peripheral_connection=(blue_controller, 35)
  quit
install_physical_configuration
quit
```

If you have a 10395-11 (836), FA7B4-D (836), 7155-1 or 7155-1x controller connected to more than one channel and the channel or the 7255-1 channel adapter fails, you may logically reconfigure NOS/VE to use one of the remaining channels, as follows. NOS/VE uses only the first channel defined in the controller's IOU_CONNECTION parameter. Change the order of the channels in the IOU_CONNECTION parameter to get NOS/VE to use the channel you wish. For example, an FA7B4-D controller originally was

defined as follows when channel 6 failed:

```
define_element element=red_cml ei=$fa7b4_d ..  
sn=1245 ic=((ch6 1) (ch0 1))
```

To reconfigure to channel 0, do the following:

```
edit_physical_configuration  
change_element_definition element=red_cml ..  
ic = ((ch0 1) (ch6 1))  
quit  
install_physical_configuration  
quit
```

5. Enter NO or GO to subsequent operator pauses.
6. Complete the deadstart.
7. Request a CE to diagnose and repair the problem while you are using the system.
8. If you need to recable after the repairs are completed, you must perform this repair solution again starting at step 1 when it is convenient for you to do so. After repair, if you need to reinstate volumes which are now accessible, you must perform repair solution 2 when it is convenient for you to do so.

F.6 Repair Solution 4

This solution deals with:

1. A failure on the NOS/VE system disk which could not be immediately repaired and which, therefore, requires a new system disk to be chosen.
2. Any hardware or software failure which causes the loss of the entire file base.

Repair solution:

1. Perform a NOS/VE installation deadstart with the current release materials. An installation deadstart requires and presumes deadstart pause for operator input. Note that if you are operating a standalone NOS/VE system and the CIP device must be recabled or an alternate CIP device is to be used, you must change the deadstart panel setting to describe the new location of the CIP device. Refer to the CIP User's Handbook to determine what information must be specified. If the CIP device is also the NOS/VE system disk, you will also need to describe the reconfiguration of the system disk to NOS/VE. For more information on that subject, refer to the section called Reconfiguration of NOS/VE Deadstart Peripherals later in this appendix.

In a dual-state system, you can only reconfigure the CIP device when NOS or NOS/BE is deadstarted.

2. Use the PCU to change the state of failing storage device(s) or controller(s) to DOWN. If you have stand-by mass storage channels, controllers or storage devices you wish to introduce to the configuration to substitute for those to be repaired, then either change their state to ON (if they are currently described in your physical configuration) or describe them now, using the EDIPC ADD_ELEMENT_DEFINITION subcommand.

When the following prompt appears:

```
Do you want to change the mainframe's physical configuration?  
Enter yes or no:
```

answer YES or Y. Then enter:

```
edit_physical_configuration  
  change_element_definition element=<name> state=down  
quit  
install_physical_configuration  
quit
```

3. Use the LCU to:

- a. Change the state of a failing channel to DOWN using the CHANGE_ELEMENT_STATE subcommand.

- b. Initialize mass storage volumes and add them to the set. When the following prompt appears:

Do you want to change the mainframe's logical configuration?
Enter yes or no

answer YES or Y. Then enter:

```
change_element_state element=ch4 state=down
```

After you enter the `CHANGE_ELEMENT_STATE` subcommand, enter an `INITIALIZE_MS_VOLUME` and an `ADD_VOLUME_TO_SET` subcommand for each volume to be initialized:

```
initialize_ms_volume element=<name> recorded_vsn=<name>  
add_volume_to_set member_vsn=<name>
```

When you are finished, type `QUIT` to exit the LCU:

```
quit
```

- c. Complete the installation deadstart.
- d. Use the `$SYSTEM.RESTORE` command to restore all the partial back-up tapes in order of most recent to least recent and then to restore the last full back-up tapes:

```
attach file $system.restore  
restore last_backup=true ..  
    backup_vsn count=6 ..  
    type=mt9$6250
```

- e. Request a CE to diagnose and repair the problem while you are using the system.
- f. If a disk storage device has been repaired and you want to reinstate it, perform the following steps when convenient. It is not necessary to perform a continuation deadstart to reinstate any repaired channel, controller or tape storage device; simply use the LCU subcommand `CHANGE_ELEMENT_STATE`. See repair solution 5 for an example.

Enter the `TERMINATE_SYSTEM` operator command.

Perform a NOS/VE continuation deadstart requesting deadstart pause for operator input.

Enter `GO` (rather than `AUTO`) to terminate system core command processing.

Use the PCU to change the state of each repaired storage device back to ON and to return temporarily used devices back to their "stand-by" (OFF) state. Because you performed an installation deadstart, the

files and catalogs on the repaired devices are not recoverable; therefore, you must initialize each reinstated volume. The following dialogue shows the sequence of PCU and LCU commands you must enter to reinstate the repaired devices. When the following prompt appears:

Do you want to change the mainframe's physical configuration?
Enter yes or no:

answer YES or Y. Then enter:

```
edit_physical_configuration
  change_element_definition element=<name> state=on
quit
install_physical_configuration
quit
```

When the following prompt appears:

Do you want to change the mainframe's logical configuration?
Enter yes or no

answer YES or Y. Then enter the following pair of subcommands for each storage device to be reinstated:

```
initialize_ms_volume element=<name> recorded_vsn=<name>
add_volume_to_set member_vsn=<name>
```

When you are finished, enter QUIT to exit the LCU:

```
quit
```

F.7 Repair Solution 5

This solution deals with magnetic tape controller and channel cable failures in a 7021-32 configuration where the alternate access is available and usable (that is, in the ON state).

1. Change the state of the failing channel or controller to DOWN using the LCU subcommand CHANGE_ELEMENT_STATE.

Enter:

```
lcu
  change_element_state element=<name>state=down
    -or-
  change_element_state element=ch<n> state=down
quit
```

2. Request a CE to diagnose and repair the problem while you are using the system.
3. Change the state of the repaired channel or controller to ON using the LCU subcommand CHANGE_ELEMENT_STATE.

Enter:

```
lcu
  change_element_state element=<name> state=on
    -or-
  change_element_state element=ch<n> state=down
quit
```

F.8 Repair Solution 6

This solution deals with magnetic tape controller and channel cable failures in a 7021-31 or 7221-1 configuration or in a 7021-32 configuration in which the alternate access is also DOWN.

1. Terminate all jobs which have tape storage devices assigned to them which are no longer accessible due to the failure. This allows the downing of the failing tape channel or controller. Use the VED,TAPE_STATUS display to determine which jobs have been assigned to the tape storage devices in question.
2. Follow all the steps in Repair Solution 5.

F.9 Repair Solution 7

This solution deals with magnetic tape storage device failures in either a single or dual access configuration.

1. Terminate the job or task that was using the storage device at the time of the failure. Use the VED,TAPE_STATUS display to determine which job has the tape storage device assigned to it.
2. Change the state of the failing storage device to DOWN using the LCU subcommand CHANGE_ELEMENT_STATE.

Enter:

```
lcu
  change_element_state element=<name> state=down
quit
```

3. Request a CE to diagnose and repair the problem while you are using the system.
4. Change the state of the repaired storage device to ON using the LCU subcommand CHANGE_ELEMENT_STATE.

Enter:

```
lcu
  change_element_state element=<name> state=on
quit
```


F.10 Reconfiguration of NOS/VE Deadstart Peripherals

Use the following process to reconfigure the tape or mass storage peripherals used to deadstart NOS/VE. In addition to the following steps, which describe the configuration of the NOS/VE deadstart device, you will also have to change the installed NOS/VE physical configuration to reflect the reconfiguration of the system disk or tape peripherals. Refer to repair solution 3 for an example of changing the installed configuration on a continuation deadstart. Refer to repair solution 4 for an example of changing the installed configuration on an installation deadstart.

In a standalone NOS/VE system, do the following:

1. Initiate deadstart with operator pause.
2. Deadstart the machine.
3. Select the O option in the first CIP display to change the default address of the NOS/VE deadstart file.
4. Select the S option in the next CIP display to specify the device class used to access the NOS/VE deadstart file.
5. Select the T option in the CIP display if you are doing a NOS/VE installation deadstart or a continuation deadstart from a tape device; otherwise, specify NEXT/RETURN.
6. At the next display, change or validate the device class and the address of the peripherals which are used to access the NOS/VE deadstart file.

In a dual-state NOS/VE system do the following:

1. Initiate the NOS/VE deadstart at the NOS or NOS/BE system console.
2. If you have recabled, NOS/VE will not be able to access its deadstart file and will display the deadstart boot display. On this display, enter the values that describe the new deadstart device. Also, specify deadstart pause for operator input.

Appendix G

DISPLAY_SYSTEM_DATA Performance Tool

DISPLAY_SYSTEM_DATA (DISSD) is currently the most complete system performance tool available in NOS/VE. It was created as a tool for the NOS/VE development team. As such, not all the data it contains is useful to customers for NOS/VE performance analysis. A future release will provide a different performance tool for site analysts. This tool will share many data items with DISSD, but omit those that are only useful to NOS/VE systems developers. While the temporary use of DISSD is encouraged, its data items and formats are subject to more radical change than a standardized site analysis tool will be.

This appendix provides a description of those data items most useful to a site analyst. The meaning and use of these data items should not change no matter what the reporting vehicle is.

DISSD is an SCL command that can be invoked from a user's terminal or from the system console.

DISSD parameters are:

```
display_option, ..
display_options, do      : list of key page_faults, pf, ..
                          monitor_requests, mr, ..
                          cpu_statistics, cs, ..
                          pio_statistics, ps, ..
                          aging_statistics, as, ..
                          jm_mm_statistics, jms, ..
                          swap_statistics, ss, ..
                          job_sched_statistics, jss, all = all
display_format, df       : key incremental, i, total, t = incremental
reset_maximum_time, rmt  : boolean = false
output, o                : file = $output
status                   : var of status = $optional
```

Parameters of interest are:

display_format There are two choices, incremental and total. The default, incremental, is the most useful. This tells DISSD to report accumulated data as the difference between the current value and the value of the accumulator the last time DISSD was run. This way DISSD only reports data for an "interval of interest". Thus to obtain data for an interval, you must invoke DISSD at the start of the interval just to establish a base point. Frequently, you discard the output from this call. At the conclusion of the interval, DISSD is called to report the data for that interval. This call will also establish the base point for the next interval. You do not

want to call DISSD at too frequent an interval, since the data is only available in formatted form. No compacted data format is available.

While the data reported by DISSD is system wide data, the incremental display_format base data items are kept in the calling job's address space. Thus each user can independently call DISSD without interfering with another. This also means that terminating a job that established an incremental base point will cause that base to be lost. The same user logging in later cannot access that data for incremental reporting.

display_options This allows you to select one, a list, or all possible display options. Each display option presents a separate set of performance data items. The display options and related data items are described in the following sections.

G.1 Page_Faults Display Option

The data items in this display are counts of the various types of page faults that can occur in the system. There are 15 different counts reported. Of these, usually only 3 are significant. They are described later. If any of the other 12 page fault counts are large, this indicates some unusual status in the system.

G.1.1 Available Queue

A page fault from the available queue is also referred to as a page reclaim. A job must have created or brought the page in from disk before it can page fault from the available queue. When a job page faults, the available queue is the most desirable place to satisfy the page fault from. The available queue consists of pages that have been removed from user jobs but have not been claimed for other uses. If a job references one of these pages before they are claimed, the system can give this page back to the job. This is a fast page fault. Normal values for this count are dependent on workload characteristics. However, if the value of this count shrinks in relation to page faults from disk, it may be an indication that the system is becoming constrained by memory size. The reason being that high memory demand will cause pages to be claimed by other users before being reclaimed. Thus, the page fault must be satisfied from disk.

G.1.2 On Disk

A page fault from disk is an expensive page fault. Page faults from disk are also referred to as page ins. These must be done the first time an existing page is referenced, or for references to a page that has been aged out and deleted from memory. If the relative count of page faults from disk to page faults from the available queue grows, this is an indication of changing workload characteristics or an increase in workload that is resulting in more memory contention.

System and job class attributes that can have an effect on page faults from disk and page faults from the available queue include:

1. PAGE_AGING_INTERVAL
2. JOB_WORKING_SET_AGE_INTERVAL
3. SHARED_WORKING_SET_AGE_INTERVAL
4. SWAPPING_AIC
5. AGGRESSIVE_AGING_LEVEL
6. AGGRESSIVE_AGING_LEVEL_2

You should not change these values from system defaults unless you have a very good understanding of what you are doing.

G.1.3 New Pages

As the title suggests, this is a count of the number of pages newly created by NOS/VE. New page page faults are also called page assigns. This can be the result of file creation or extension. This is strictly workload dependent. It does not include modification of previously existing pages.

G.1.4 Available Modified Queue

This is a count of the number of page faults that were satisfied from the available modified queue. Pages are placed in the available modified queue if they have been aged out of a job's working set and were modified. The system will attempt to write an available modified page to disk as soon as possible. When this is done the page will be placed in the available queue. Thus it is most common for the count of page faults satisfied from the available modified queue to be small compared to the number of page faults satisfied from the available queue.

G.1.5 No Memory

The NOS/VE System Performance and Maintenance Manual (60463915) describes the meaning of this data item and also what action to take when this item is too large. This description can be found in the description of the system attribute AGGRESSIVE_AGING_LEVEL.

G.1.6 Low on Memory

The NOS/VE System Performance and Maintenance Manual (60463915) describes the meaning of this data item and also what action to take when this item is too large. This description can be found in the description of the system attribute AGGRESSIVE_AGING_LEVEL_2.

G.2 Monitor_Requests Display Option

This section reports data about requests to monitor functions. While it is most useful as an internal development tool, there are some items that can be used by a site analyst.

There are seven columns. The first column identifies the monitor function. Data for a monitor function is only reported if its request count is not 0. The second column is the number of times the monitor request was called. The third column is the average number of microseconds it took to execute the monitor function. The fourth column reports the maximum time spent satisfying a single request. The fifth column is the percent of CPU time spent in the monitor function relative to all the other monitor functions. The sixth column is the number of monitor requests divided by the DISSD time increment. The seventh column is the total number of CPU seconds used by the named monitor request.

The column of most interest is the percent time column. This will allow you to quickly identify which requests took the most time. The requests that typically stand out are:

1. page_fault
2. write_modified_pages
3. switch_task
4. process_io
5. periodic_call

While this list may be somewhat different at individual sites, the following analysis will still apply. It is worth noting if a new request stands out, or if a request that normally stands out doesn't. This alone may help account for performance changes. To see why a request's time percentage has changed, look at the count to see if it is being called more. Look at the average request time to see if the average time to satisfy a request has changed. To understand if an average time change is a trend or the result of a single request, look at the maximum time for the request. If it is in line, then the change is probably the result of a trend. If it is not, then the change could just be the result of a single request.

For dual CPU systems, there are times when you will see a second set of data for monitor requests. This reports time spent queueing for a monitor request. This section has four columns: request name, number of times queued for the request, average time spent waiting in microseconds, and maximum time spent waiting in microseconds. If these counts are too high, this is a good indication of CPU queueing.

G.3 Aging_Statistics Display Option

This section reports counts of the times the listed aging functions occur. These functions can be broken into three classes.

aggressive aging	It is normal for these counts to be very small. If not, you may have a memory contention problem.
normal aging functions	This is typically not of interest. A description of NOS/VE paging is in the NOS/VE System Performance and Maintenance Manual (60463915).
write page functions	These are normally small and are typically not of interest.

G.4 JM_MM_Statistics Display Option

This section provides a snapshot of the reported data items at the instant the command was executed. It is sampled data rather than accumulated data (i.e. - it is not a running average). To obtain a more accurate picture of the behavior of the reported data items in a given time interval, one would have to take several samples. The frequency of this sampling must be balanced against the resulting system overhead.

The first grouping of data reports the sizes of the various page queues maintained by NOS/VE. These are actual pages in memory. Thus this data shows how physical memory is being used. There are two other page queue whose size is not reported here. They are the swap resident queue and the long wait queue. Pages in these queues are from jobs that have been swapped out, but whose memory image is still intact. Because these two queue's pages are not reported one must be careful in using the total number of pages reported in this section. If there are pages in these queues the total number of pages reported will be less than the actual number of pages in real memory. If the only job in the system is the system job, then the reported count will be correct.

1. free queue

Pages in this queue are not associated with any virtual address. They can be claimed at any time by any job. The free queue is usually small during busy times. This is normal and should not cause concern.

2. available queue

Pages in the available queue are pages that have been aged out of a job's working set and are available to be claimed by any job. However the system maintains the page's association with the job it was aged out of. If this job page faults for the page before another job claims it, the page is put back in the job's working set. These are inexpensive page

faults. This is often one of the larger page queues.

3. available modified queue

This queue functions just like the available queue except pages in this queue have been modified before being aged out. Thus they must be written to disk before they can be claimed by another job. The system attempts to write the page back to disk as soon as possible. Once this is accomplished, the page is moved to the available queue. Because pages are quickly moved out of the available modified queue into the available queue, the size of the available modified queue is usually small.

4. wired queue

This queue consists of all pages that cannot be paged or swapped out of memory. The size of this queue stays relatively constant, but it can change during the course of a day. The pages in this queue are generally important operating system tables.

5. shared queue

The pages in this queue are pages that have actually been used by more than one job. One must make the distinction between sharable and shared pages. Sharable pages exist in a job's working set until they are shared. Pages sharable for write will always be in the shared queue. Almost all operating system code is sharable. In normal situations, this will make up the bulk of shared pages. Pages in the shared queue are aged out of this queue in a manner analogous to pages in a job's working set. However because these pages benefit more than one job, they have a separate aging interval which is, by default, longer than that for a job's working set.

6. job fixed queue

Pages in this queue are swappable but not pageable. They are pages that are needed when a job is in memory, but are not needed by the system when that job is not in memory.

7. job io error queue

This queue should have no pages in it.

8. job working set queue

These are pageable, swappable pages that are being used by a single job.

The next set of data reports the number of jobs in the system subtotaled by job type. It distinguishes three job types: System, Interactive, and Noninteractive. The system job is counted as a system job and as a noninteractive job.

The next block of four data items summarizes the memory residence of the jobs in the system. No attempt is made to report this data by job class.

1. Number of active jobs

This is a count of the number of jobs actually in memory

2. Number of jobs swapping in

3. Number of jobs swapping out

4. Number of jobs swapped out

The next data item reports the number of tasks that are in memory that are ready to execute (Number of ready/executing tasks)

The final data item reports the number of tasks that have come ready, but are swapped out (Number of ready but swapped tasks). This is most likely the result of the expiration of a timed wait.

G.5 Swap_Statistics Display Option

Swapping data can be useful in determining whether or not a system is memory bound. Since swapping I/O can dominate a system's disk I/O, swapping data is also useful for understanding a system's I/O load.

Many of the data items in this display are for the NOS/VE development team. The items mentioned here are the ones most important for site performance analysis.

The data is presented with five column headings. The first column heading is TRANSITION. The items under this are arranged in two columns of one or two letter mnemonics. These mnemonics describe intermediate states in the process of swapping a job in or out. The most important of these swap state transitions are discussed later. The second column is a count of the number of times the labelled swap state transition occurred. Only transitions with nonzero counts are reported. The third column is the average number of microseconds the reported swap state transition took. The fourth column is the number of transitions that occurred divided by the DISSD time increment. The fifth column is the maximum time taken by any swap state transition of the reported kind.

The following are important swap state transitions that are useful for site performance analysis.

1. R --> OR

This transition is the transition from ready (R) to swapout requested (OR). This counts the number of job scheduler initiated swapouts. This generally represents a forced swapout due to memory contention. The swapout should cause disk I/O.

2. R --> TC

This is the transition from executing (R) to idle tasks complete (TC). For interactive intensive workloads this is the most common swapout. It is caused by an interactive job going into "long wait" waiting for a response from the user at a terminal. This does not necessarily cause disk I/O to occur. Whether or not the job is swapped out to disk depends on the degree of memory contention in the system.

3. OI --> OC

This count is the actual number of job swapouts that occurred. This count will be different from the swapout requests reported in the PIO disk unit data. This is because swap files generally exceed the maximum disk transfer size. Therefore the system job swapper must break each swap file into chunks and make separate requests to the I/O procedures to transfer each chunk.

4. S --> IR

This transition is the count of swapin requests that have to be satisfied from disk. If this number is a large percentage of the sum of the S --> IR and IR --> R counts, it is possible the system is memory bound and could benefit from more memory.

5. IR --> R

This is the count of swapin requests that do not require disk I/O. These swapin requests are satisfied from the long wait queue (no disk I/O has occurred) or the swap resident queue (swapout I/O was done, but job memory image pages were not claimed by another job). The sum of the S --> IR and IR --> R transitions is the total number of swapins for the system.

6. II --> IC

This is the transition from swapin I/O initiated to swapin I/O completed. This count is the actual number of swapins that occurred from disk. As in the case of swapouts to disk, each swapin must be broken into more than one swapping I/O request to the I/O system to transfer the whole swap file. Thus the Swapin Request count from the PIO statistics will be larger than this count.

Another interesting data item to watch in this transition is the average microseconds column. This monitors how long it takes for the swapin disk I/O. An increase in this number indicates increased I/O queueing.

The other data item of interest in this display is the average swap file size. This should give an indication of the amount of I/O required to swap a job out. The number that is reported is the number of pages.

G.6 CPU_Statistics Display Option

The CPU statistics report idle time for each CPU in the system. The idle time is reported both as a number of microseconds and as a percentage of the total interval time. Idle time is accumulated in one of three counters:

idle with io active	This means the CPU is idle but the system is doing disk I/O that a task is waiting for.
idle without io active	The system is truly idle. No CPU activity, no disk activity.
average idle time	This tracks how long is the system idle when it is idle. Was the CPU idle due to one long burst or was it many short bursts?

G.7 PIO_Statistics Display Option

The PIO statistics report data about the I/O system at three levels; the system wide level, the channel level, and the disk unit level. In all cases, the analyst must keep in mind that this is a software view of the I/O system which only attempts to describe the world beyond the PP as channels and disks. Controllers and their delays will be included in disk or channel times. Thus one must consider the possibility that an item such as seek and latency time for a disk also includes controller contention or delays.

The channel level data includes nine items. Each row contains data for a single channel. First is the channel number. Second is the channel utilization. A 50% channel utilization is represented by '50.0'. The channel utilization is not the time the channel is actually busy transferring data. Rather, it is the time the software has the channel reserved. Next, the total number of requests on the channel is reported. The next column is the MAU's read (from disk to memory). In the section on configuration management statistics, the NOS/VE System Performance and Maintenance Manual (60463915) describes how many bytes are in an MAU for each type of disk device. In general an MAU is 2048 bytes. The column heading "Mau's Written Data and Preset" reports the number of MAU's transferred from memory to disk. The data in the "Time per Request" column reports the average channel time taken to satisfy a request. This does not include queueing time. The "Average Q Size" is the number of requests in the channel queue. This includes the request currently being processed. The next item is a count of the recovered errors on the channel. This should generally be zero. If it is not, this could be the cause of performance problems. The last column is a count of the unrecovered errors on the channel. Again, this could be a cause of performance problems if the reported value is nonzero.

The rows below the channel data aid in the analysis of the above data items. The first row is a sum of data for all the reported disk channels. Only columns whose sum would have meaning are summed. The next row, labelled

"/sec" is the sum per second of the data in the corresponding column. The time period used is the PIO statistics time increment. This provides a standard type of measure for comparing data from different time increments. Again only those columns where this calculation makes sense are reported. The next row is an average over all disk channels for the appropriate column. The final row is the average per second for the PIO statistic time increment.

The next set of data is for the disk level. First, the disk unit number is identified. The next five columns all report counts for various types of disk requests, ending with a total count of requests. Two points are of interest here. First, the swapin and swapout counts are maintained separately from the read and write requests. They are not a subset of the read and write requests. Second, the swapping request counts are not the number of swaps that occurred, but rather the number of I/O system swapping requests required to complete swaps. See the description of Swap_Statistics for an explanation of this. You should expect to see more swapout requests than swapin requests. This is because the swap resident feature should allow the system to satisfy the swapin from swap resident and avoid the swapin I/O. When the swapin and swapout counts are close, the system may be memory bound. The final column in the first grouping of disk statistics is a count of the total number of cylinders skipped for all seeks on the disk. This will have an effect on the seek time for the disk.

The rows under this grouping are similar to those for the channel data. Two new rows have been added "/req" and "/sk". The "/req" row divides the sum of data items in the column by the total number of disk requests. For the request count columns, this provides the ratio of each request type per total requests. This can be converted to a percentage by multiplying by 100. The "/sk" row divides the sum of data items in the column by the total number of seeks.

The next grouping of disk unit data starts out with descriptions of the amount of data transferred by each disk. This is broken down by data read and data written. The "Mau's Written Data" is a subset of the "Mau's Written Data and Preset". The total MAU's transferred is the sum of MAU's read and MAU's written data and preset. The final two columns report the number of errors that occurred on each disk. If these counts are not zero, this may be the cause of performance problems.

The rows below this grouping are the same as in the previous grouping with one exception. A "pct" row has been added. This is used to report the ratio of MAU's transferred by transfer type to the total MAU's transferred. The reported numbers are ratios. Thus, a value of .50 means 50% of all MAU's transferred were of that data transfer type.

The next grouping is the system level I/O statistics. Three items are of particular significance. First, the number of seeks is reported. This will always be less than the total number of requests, but is usually not a lot less. Second, the average seek and latency time for all seeks is reported. Remember this will include controller delays. This value is usually less than the nominal disk unit seek and latency time because of seek overlap. Finally, the average I/O request wait time in queue is reported. This gives a direct measure of the impact I/O queueing has on performance.

G.8 Job_Sched_Statistics Display Option

There are two categories of useful data in this section: memory queueing data and AJL ordinal queueing data. The row labelled 'memory wait - no preempt' gives a count of the number of times a job needed to wait for memory to free up before it could be swapped in. This, along with the swap state transition data is a good indicator of the degree of memory contention in the system. The row labelled 'wait for ajlo - no preempt' gives a count of the number of times a job needed to wait for an entry in the AJL to free up before it could be swapped in. If you do not wish to use the number of active jobs allowed in the system to control the workload on the system, you may need to increase the number of active jobs allowed if this count is nonzero or larger than you wish it to be.

Appendix H
VED IJD Operator Display

The display produced by the operator command:

VEDISPLAY DO=INITIATED_JOBS_DETAILED

displays detailed status information for all initiated jobs (both swapped and nonswapped). The following fields are displayed for each job:

ordinal.ujn cpm cpj s/r ws rt srvss pr ses

ordinal Ordinal number of the job. Ordinal zero is always the system job.

ujn First 8 characters of the user-supplied name of the job.

cpm Number of central processor seconds the job has used in monitor mode.

cjm Number of central processor seconds the job has used in job mode.

s/r Job status indicator - jobs are either in memory, swapped, swapping, or have an error status.

If the job is in memory, it has a status of R.

If the job is swapped out, the following swap reasons can be displayed:

RJ job requested
RO operator requested
RT system thrashing
RP lower priority
RI idling system swapout
RW long wait
RH hung task in job

If the job is currently swapping, then the swap status (either swapin or swapout) is displayed. The swap statuses for swapout are:

OR Swapout has been requested.
TI Job is waiting for tasks to idle.
TC All tasks have been idled.
DW Wait to allocate the swap file descriptor.
DA Swap file descriptor allocated.
AJ Job mode allocate of swap file.

AW Wait to allocate swap file.
AC Swap file allocated.
JW Job is waiting for outstanding I/O to complete.
JC Job I/O is complete.
OS Initiate swapout I/O.
OW Wait to initiate swapout I/O.
OI Swapout I/O has been initiated. Waiting for swap I/O to complete.
OC Swapout I/O complete.
FM Free swapped job's memory.

The swap statuses for swapin are:

IR Swapin has been requested.
IS Swapin tables have been assigned. I/O is being initiated.
IW Swapin requested; wait to initiate swapin I/O.
II Swapin I/O has been initiated. Waiting for swap I/O to complete.
IC Swapin I/O has completed. Job tables being rebuilt.

NOTE

Contact your site analyst if a job consistently stays in one of these swap statuses.

The possible error statuses for a job are:

*R Job was damaged during recovery. The rest of the operating system is functioning normally. The only way to remove this job is to deadstart NOS/VE again.
*H Job is halted but the system is unable to terminate the job. The rest of the operating system is functioning normally. The only way to remove this job is to deadstart NOS/VE again.

NOTE

You should contact a site analyst if a job is in the *R or *H state.

ws If the job is in memory, the number of pages in the job's working set. If the job is swapped, the swapped job page count.

rt If the job is in memory, the number of tasks the job has ready for execution. If the job is swapped, and has ready tasks, the number of seconds the job has been waiting to be swapped in.

srvss Service given since last swap.

pr If the job is in memory, the job's priority. If the job is swapped, the job's swap queue priority.

ses If the job is in memory, this field is blank. If the job is swapped, the job's swap entry status, displayed as one of the following:

- NS not swapped
- OI operator force-in
- SC swapin candidate
- SI swapin initiated
- OO operator force-out
- SE system event (long wait)
- SF system force-out

Appendix I Common Site Analyst Questions

There are questions about NOS/VE asked by site analysts more frequently than others. This appendix includes answers to these questions.

1. How can I direct subsequent output files to print on the host side (NOS or NOS/BE) of dual state? At the public I/O station on the NOS/VE side?

To print on the host side use the following operator command:

```
CHANGE_JOB_ATTRIBUTE_DEFAULTS DESTINATION_USAGE=DUAL_STATE
```

To print on the NOS/VE side at a public I/O station use the following command:

```
CHANGE_JOB_ATTRIBUTE_DEFAULTS DESTINATION_USAGE=PUBLIC ..  
STATION=station-name
```

where "station-name" is that value for the parameter, I_O_STATION_NAME, on the DEFINE_I_O_STATION command in the Terminal Definition Procedure(s) for the batch output device(s).

Use of the above command changes the default job attributes for all subsequent user jobs. Users can override these defaults with a variety of commands (e.g., CHAJA, CHAOA, PRIF). For more detailed information on the CHAJAD and DEFIOS commands, consult the NOS/VE Operations Manual (60463914) and the CDCNET Site Administration and Configuration Manual (60461550).

To have the CHANGE_JOB_ATTRIBUTE_DEFAULTS command be effective at every deadstart, include it in your start up commands file.

2. Why can't my users interactively login to NOS/VE?

The most common reason is that the system default family name differs from the name of the family for which users are validated. The system default family name is the login_family that is displayed via the system operator command, DISPLAY_JOB_ATTRIBUTE_DEFAULTS.

Users logging in thru NAM/VE can specify their family name during the login dialogue.

Users logging in thru dual state do not have this option, so the family they are validated for must be the system default family. The system default family name can be changed through the LOGIN_FAMILY parameter of the CHANGE_JOB_ATTRIBUTE_DEFAULT command.

One other reason users may not be able to login thru NOS dual state is that they may not be validated for the application, VEIAF.

For more detail, consult the NOS/VE Operations Manual (60463914) and the Family Administration Manual (60464513).

3. Why won't output files print at the public I/O station?

The most common reasons are either:

- a. The value of the output file attribute, STATION, doesn't match that of the value of the I_O_STATION_NAME parameter on the DEFINE_I_O_STATION command in the Terminal Definition Procedure(s).
- b. The value of the output file attribute, DESTINATION_USAGE, is not PUBLIC.

Usually, both of the above situations can be corrected by changing the system defaults for STATION or DESTINATION_USAGE. This can be done via the operator command, CHANGE_JOB_ATTRIBUTE_DEFAULTS.

Each batch device has attributes that affect file selection. If any of the following attributes do not match the attributes for the output file, it will not be selected for printing.

1. EXTERNAL_CHARACTERISTICS
2. FORMS_CODE
3. MAXIMUM_FILE_SIZE
4. PAGE_LENGTH
5. PAGE_WIDTH

For detailed information, consult the NOS/VE Operations Manual (60463914) and the CDCNET Batch Device User Guide Manual (60463863).

Appendix J

NOS/VE REDO Facility

The NOS/VE REDO facility allows you to modify and execute a previously entered command without retyping the entire command.

NOTE

This capability is not supported with NOS/VE 1.2.1 L670, but is provided for customer use and experimentation. Feedback is welcome but formal PSRs will not be answered. Send comments to Control Data Corporation, Attention: Richard Ragan, 215 Moffett Park Drive, Sunnyvale, CA 94087

To use REDO add the following to your user prolog:

```
CHANGE_TERMINAL_ATTRIBUTES tm=terminal-model-name
$SYSTEM.OSF$SITE_COMMAND_LIBRARY.ENABLE_COMMAND_REDO
```

where "terminal-model-name" is the value for the type of terminal you are using. These two commands must be in your user prolog. REDO will not work correctly if you do not include these commands in your user prolog.

All NOS/VE commands you enter from the keyboard are now saved, and you can search for and edit any commands you previously entered. You can also define frequently entered commands to be associated with function keys. Refer to 'Associating Function Keys with Commands' below.

Activation of REDO depends on how your terminal has been defined. If your terminal definition includes a BKW and FWD key definition, then refer to the next section. Otherwise skip to the section entitled 'Activating REDO without BKW/FWD Keys'.

J.1 Activating REDO with BKW/FWD Keys

If you want to edit a previously typed command, press the key you use for the BKW (backward) function in EDIT_FILE. The last command you entered at the terminal will appear.

Use the BKW and FWD function keys to display commands prior (BKW) or subsequent (FWD) to the currently displayed command.

You can locate a specific command by typing the first few characters of the command and then pressing BKW. REDO will search backwards for the first command starting with that string and display it.

J.2 Activating REDO without BKW/FWD Keys

If your terminal does not have a BKW function key defined for it, enter the up cursor key and a carriage return to activate REDO. The last command you entered at the terminal will appear.

Alternatively, you can locate a specific command by typing the first few characters of the command and then the up cursor key and a carriage return. REDO will search backwards for the first command starting with that string and display it.

Once you've activated REDO, you can display commands prior to the currently displayed command, with the up cursor key (no carriage return key is necessary).

To display commands following the currently displayed command, use the down cursor key (no carriage return key is necessary).

J.3 Editing Commands with REDO

Once you find the command of interest, you can edit it using the same techniques as you do in EDIT_FILE. You can move the cursor right and left with the cursor keys, use insert and delete character, overstrike characters, and, if your terminal supports it, clear to end of line. When you are ready to execute the command, just press carriage return. If you decide you want to get out of this mode, there are several choices:

1. press the BKW function key
2. press the FWD function key until you get the NOS/VE command prompt back
3. clear the current line to blanks and press carriage return
4. enter CONTROL/T

The UP function key will position the cursor to the end of the command. The DOWN function key will position the cursor to the beginning of the command. CONTROL/D will delete the "word" to the left of the cursor. Delete word works as follows:

1. If a non-SCL-name character is to the left of the cursor, delete such characters until an SCL-name character is encountered; then delete SCL-name characters until a non-SCL name character is found.
2. If an SCL-name character is to the left of the cursor, delete such characters until a non-SCL-name character is found.

If you enter continued SCL commands (i.e., ending in ..), REDO will concatenate all the parts into a single long command. If the command is wider than your screen (REDO uses your terminal attribute PAGE_WIDTH setting to determine your screen width) only the part that will fit is displayed. To see parts of the command that are off the screen, move the cursor to the end of the line (usually column 80) and press one more cursor right. The command line will then be shifted twenty characters to the left and the cursor placed

in logical column 81. Repeating this process will shift the line by 20 each time. The reverse process works to shift the line to the right (cursor left from column one). Typing off the end of the line will automatically shift the line too. Deleting characters from a wide line will repaint the character entering the screen on the right after each delete character press. Depending on NOS/VE response time, the actual refresh may lag behind your key pressing. If you do this, it is advisable to verify that "what you see is what you get" because half duplex mode can cause problems if output comes out while you are typing in. Pressing any control key not recognized by REDO will refresh the current line. CONTROL/R is a good mnemonic one to use (R for refresh).

J.4 Associating Function Keys with Commands

REDO also supports unshifted and shifted function key strings. Just define SCL variables of the form Fn and SFn (e.g, f1, sf1, f3, sf7, etc.) to correspond with the function keys and shifted function keys. Thus to make the F1 key do a DISPLAY_CATALOG, you would enter:

```
F1='DISPLAY_CATALOG'
```

Now when you press the F1 function key, REDO will issue the string found in the variable F1. If the function key string ends in two periods (..), the key will be transmitted but no carriage return will be done. This lets you define a key such as

```
SF1='FTN,L=LIST,OPT=DEBUG,DA=ALL,B=BIN,I=..'
```

and then enter shifted function key 1, followed by the value for the I parameter.

Function keys may also be pressed while in REDO editing mode. The string defined for the key will be inserted to the left of the cursor position. The presence or absence of an ending .. then determines whether the command is executed or more editing takes place.

J.5 Full Duplex Mode

For terminals lacking insert/delete character functions, REDO supports a true full duplex mode. In order to use this you must set the terminal attribute ECHOPLEX to yes and invoke REDO with

```
$SYSTEM.OSF$SITE_COMMAND_LIBRARY.ENABLE_COMMAND_REDO ..  
FULL_DUPLEX=YES
```

in your user prolog.

If you want your default insert mode to insert characters instead of overstriking them, add the parameter `insert_mode=yes` to the `ENABLE_COMMAND_REDO` command.

CONTROL/A will toggle between insert mode and overstrike mode while editing in REDO. While in REDO, each character you type is echoed to the terminal all the way from REDO in NOS/VE (not by the DI or 2550). This can result in a lag between your typing and the appearance of the characters on the screen. However, the screen should always faithfully represent what REDO has seen.

J.6 REDO Command Summary

KEY SEQUENCE	DESCRIPTION
BKW or Up Arrow	Initiate REDO/Back up one command (*)
FWD or Down Arrow	Forward one command (*)
UP	Cursor to end of command
DOWN	Cursor to start of command
Left/Right Arrow	Cursor Left/Right, shifts line 20 characters if cursor goes off left or right edge of screen
Insert Character	Inserts a blank character (if terminal has this)
Delete Character	Deletes a character (if terminal has this)
Insert On/Off	Supports terminals with insert on/off modes
CONTROL/D	Delete Word to left of cursor
CONTROL/U	Deletes from column 1 up to cursor position
CONTROL/A	Toggles software insert character mode on/off
CONTROL/R	Refresh current line
Clear to end of line	Supported if terminal has this
Erase character	Supported if terminal has this
Backspace	Same as left arrow
Del	Delete character to left of cursor
BACK, CONTROL/T, STOP	Exits from REDO to SCL command mode
F1-Fn, SF1-SFn	Executes string in SCL variable F1-Fn, SF1-SFn (**)

(*) - Cursor alone must be followed by a CR to activate REDO.

(**) - Function key strings ending in .. allow further editing after the function key is inserted. If in REDO mode already, the F-key string is inserted to the left of the cursor.

J.7 Significant Problems

The following are significant problems with the REDO processor.

1. REDO does not work well on a CDC 721 if you have used the NOS KEY utility to load your function keys. To clear out any NOS KEY definitions, press SETUP, F10, F10, F1. Then things should work better.

2. REDO runs in single character forwarding transparent mode. Using NAM/CCP, especially with ANSI X3.64 type terminals e.g. VT100, the 63 character buffer in the 2550 is easily overrun if the cursor keys are held down in repeat mode. This will result in messages such as WAIT.. and REPEAT.. overwriting the line being redone. CONTROL/R will refresh the line and reestablish the correct cursor position on the screen. Each VT100 arrow key press sends 3 characters thus 21 arrow keypresses can fill the 2550 buffer and repeat mode aggravates the problem. Running with some sort of flow control between the terminal and the 2550 should eliminate the problem. Otherwise, no workaround is known or planned. The problem does not exist with CDCNET and DI's.
3. If REDO is active, file transfers using MacConnect with the CONNECT protocol do not work. Symptoms are that the file transfer cannot be completed.
4. Commands started with REDO cannot be aborted by a terminate break unless they have written some output to the terminal.
5. Typed-ahead input will sometimes be incorrectly handled.
6. Users of VT100 compatible terminals may wish to modify their TDU file to improve REDO usability with the function key feature. VT100 function keys are defined as the alternate keysequences of the numeric keypad. When the terminal is in line mode (as opposed to screen mode), the current VT100 TDU file released by CDC leaves the numeric pad operating as a numeric keypad. This means that keypad presses (other than PF1, PF2, PF3, PF4) will be seen as number keys and REDO will not treat them as function keys. You may want to change the TDU file for the VT100 so that the Set Line Mode sequence of codes will leave the numeric pad in alternate keypad mode. This will allow the full set of keys on the numeric pad to be interpreted as function keys. Removing the sequence 1B(16) 1E(16) which resets the numeric pad to number mode will accomplish this change.

Appendix K Online EXAMPLES Manual

The EXAMPLES online manual accesses examples illustrating various NOS/VE topics, categorized by NOS/VE printed manual names. You can access the online manual by entering

EXPLAIN m=examples

The following describes how to use the EXAMPLES online manual to:

1. view an example
2. copy an example
3. print an example
4. access an example by name
5. try out an example

K.1 Viewing An Example

Suppose you want to see and experiment with an example of the CLP\$CREATE_VARIABLE interface which is documented in the printed manual CYBIL System Interface.

1. Entering

top

results in a menu like the following:

- a. CYBIL File Management
- b. CYBIL Keyed-File and Sort/Merge Interfaces
- c. CYBIL System Interface
- d. CYBIL Sequential and Byte Addressable
- e. Source Code Management

2. Choose

c

to select examples of topics documented in the CYBIL System Interface Manual (60464115).

3. A menu of those examples will be listed. The menu could look like this:

- a. CLP\$CREATE_VARIABLE_1 - create SCL variables
- b. CLP\$READ_VARIABLE_1 - read value in an SCL variable
- c. CLP\$SCAN_EXPRESSION_1 - scan integer and string expressions
- d. COMMAND_PROCESSOR_EXAMPLE_1 - deletes a list of SCL variables
- e. COMMAND_UTILITY_1 - INFO_PLEASE_UTILITY from Chapter 9

4. Select

a

which is the choice for an example of the CLP\$CREATE_VARIABLE interface.

5. A description of the example and the following choices will be displayed.

- a. view the example
- b. copy the example
- c. print the example

6. Select

a

to view the example with EDIT_FILE. When you quit EDIT_FILE, you will return to the display that lists a description of the example.

K.2 Copying An Example

Suppose you want to copy an example of the CLP\$CREATE_VARIABLE interface which is documented in the printed manual CYBIL System Interface Manual (60464115).

1. Entering

top

results in a menu like the following:

- a. CYBIL File Management
- b. CYBIL Keyed-File and Sort/Merge Interfaces
- c. CYBIL System Interface
- d. CYBIL Sequential and Byte Addressable
- e. Source Code Management

2. Choose

c

to select the CYBIL System Interface Manual (60464115) manual examples.

3. A menu of all those examples will be listed. The menu could look like this:

- a. CLP\$CREATE_VARIABLE_1 - create SCL variables
- b. CLP\$READ_VARIABLE_1 - read value in an SCL variable
- c. CLP\$SCAN_EXPRESSION_1 - scan integer and string expressions
- d. COMMAND_PROCESSOR_EXAMPLE_1 - deletes a list of SCL variables
- e. COMMAND_UTILITY_1 - INFO_PLEASE_UTILITY from Chapter 9

4. Select

a

which is the choice for an example of the CLP\$CREATE_VARIABLE interface.

5. A description of the example and the following choices will be displayed.

- a. view the example
- b. copy the example
- c. print the example

6. Choose

b

to copy the example to a file. You will be asked

Name of the file to copy example to?

and should respond with the name of the file to contain a copy of the example.

For example, responding

\$user.cre_var_example

will result in the example being copied to file \$USER.CRE_VAR_EXAMPLE.

7. Leave the online manual by entering

QUIT

8. You can then use the copy of the example on file \$USER.CRE_VAR_EXAMPLE, to execute the example, experiment with it, etc.

K.3 Printing An Example

You can print any example in the EXAMPLES online manual. Suppose you wanted to print an example that demonstrates a keyed file interface.

1. Entering

top

results in a menu like the following:

- a. CYBIL File Management
- b. CYBIL Keyed-File and Sort/Merge Interfaces
- c. CYBIL System Interface
- d. CYBIL Sequential and Byte Addressable
- e. Source Code Management

Choose

b

to select the keyed file examples.

2. A menu of all the keyed file examples will be listed. The menu could look like this

- a. ALTERNATE_KEY_FILE_1 - alternate key definition
- b. CREATE_IS_FILE_1 - creates an indexed-sequential file
- c. NESTED_FILE_EXAMPLE_1 - copy and delete nested-file defs
- d. UPDATE_IS_FILE_1 - update of a keyed file

3. Select

b

which is the choice for an example of how to create an indexed-sequential file.

4. A description of the example and the following choices will be displayed.

- a. view the example
- b. copy the example
- c. print the example

5. Select

c

to print the example.

The system will first try to print the example with the command

print file-name

where "file-name" is the name of the file containing the current copy of the example.

If PRINT is not a command, then the standard NOS/VE PRINT_FILE command will be used.

This allows you or your site to define a PRINT command to control how an example is printed. If you or your site haven't defined a PRINT command, then the example will be printed using the PRINT_FILE command.

6. Leave the online manual by entering

quit

K.4 Accessing An Example by Name

The EXAMPLES online manual is indexed by all the names of the examples, the topics that the examples illustrate, and, for program interface examples, the names of the program interfaces that they exemplify.

Suppose you want to find examples of the FSP\$OPEN_FILE interface:

1. Search the index for FSP\$OPEN_FILE by entering

fsp\$open_file

this will display the first example that demonstrates FSP\$OPEN_FILE.

2. You can see all the examples that demonstrate FSP\$OPEN_FILE by entering

?

after each example description has been shown. When there are no more examples has been FSP\$OPEN_FILE, the message

Topic entries exhausted for - fsp\$open_file.

will be displayed.

You can also browse through the manual index by using the

`display_index`

command. For example, to see index entries starting with CLP\$:

1. Use the `display_index` command

`display_index clp$`

which will result in a menu list such as:

- a. `change_pf_1`
- b. `clp$convert_integer_to_rjstring`
- c. `clp$convert_integer_to_string`
- d. `clp$convert_real_to_string`
- e. `clp$convert_string_to_file`
- f. `clp$convert_string_to_real`
- g. `clp$create_variable`

2. Then you can select whichever entry is of interest.

3. Entering a

?

after the example description has been shown will result in the selection of the next example that illustrates that topic. When there are no more examples illustrating that topic, the message

Topic entries exhausted for - your-index-entry.

will be displayed, where 'your-index-entry' is the topic you were examining.

K.5 Trying Out An Example

To try out an example, copy the example to a file and then execute that file. For example, if you copied an example to file `$USER.SHOW_ME`, then after exiting from the Online Examples manual with the `QUIT` command, you could enter

`$user.show_me`

to try out the example.

Appendix L

NOS/VE Questions Most Frequently Asked

There are questions about NOS/VE usage that are asked more frequently than others. This appendix includes answers to those questions.

1. How do I do transparent input/output from a FORTRAN program?

You can do transparent input/output from FORTRAN through the CONNEC subroutine, with the character set designator set to 2 as in

```
CALL CONNEC (unit, 2)
```

The CONNEC subroutine is documented in the FORTRAN for NOS/VE Usage Manual (60485913).

2. Why are the files which I create within a FORTRAN program not displayed by DISPLAY_CATALOG?

Such files will be displayed once you have issued any SCL command specifying the file.

You can also issue an operation, such as a REWIND_FILE through the SCLCMD subroutine to make the files visible after your FORTRAN program is finished.

3. Why don't GET_FILE (GETF) or REPLACE_FILE (REPF) work on a NOS-host system?

Typical user errors include:

- a. Your NOS/VE password doesn't match your NOS batch password. You can change the VE password through the SET_PASSWORD command, and the NOS batch password by submitting a NOS batch job with the PASSWOR command in it.

NOTE

Your NOS interactive password is used when you login to NOS/VE through VEIAF.

- b. You may have defined a NOS user prolog which executed some command that aborts from a batch job. Modify your NOS user prolog by including a test for job origin, such as

```
.IF,OT.EQ.IAO,TTYORIGIN.  
... commands legal only in interactive jobs  
.ENDIF,TTYORIGIN.
```

- c. Your NOS prologs (system, project, and/or user) cannot change the file INPUT. This includes rewinding it or returning it. Be sure to check for the asterisk (*) option on any CLEAR, RETURN, REWIND, or UNLOAD commands.
- d. You must be validated for a time limit of at least 5000, as this is the released default JOB card used by GET_FILE and REPLACE_FILE.
- e. Your link attributes must be correct. In particular, if your NOS/VE family name is different from your NOS family name, then you must use the SET_LINK_ATTRIBUTES command to specify your NOS family name.
4. The STATUS variable on a command doesn't get set if the command has a syntax error such as a 'bad' filename or parameter. How can I get the status back?

SCL first checks file names and command parameters before any command is ever executed. Thus SCL, and not the command is detecting a 'bad' parameter. Use the INCLUDE_LINE command with a STATUS variable if you want to trap the status of commands.

An example of this would be to trap a command such as

```
DISPLAY_CATALOG lo=f status=cmd_status
```

which would produce

```
--ERROR-- LO is not a parameter name.
```

and not set the cmd_status variable. To trap this command's status, use

```
INCLUDE_LINE 'display_catalog lo=f' status=cmd_status
```

which would then set the cmd_status variable.

5. How can I prevent a batch job's output from being printed at the central site?

There are several options available for the disposition of output from batch jobs.

1. You can use the STANDARD_OUTPUT parameter of the SUBMIT_JOB command to direct output to a permanent file.

For example, entering:

```
SUBMIT_JOB f=my_job so=$user.my_job_output.$next
```

causes the file MY_JOB to be submitted as a batch job, and output from that job to be written to the next cycle of file \$USER.MY_JOB_OUTPUT.

2. You can use the TERMINATE_OUTPUT command, as in

```
TERMINATE_OUTPUT n=output
```

to prevent the output from being printed or written to the file specified by the STANDARD_OUTPUT parameter.

3. On dual-state systems, you can use the job attributes DESTINATION_USAGE=DUAL_STATE and DUAL_STATE_ROUTE_PARAMETERS to specify job output routing parameters. Since these are job attributes, you can use the CHANGE_JOB_ATTRIBUTE command (CHAJA) in your user prolog to automatically cause all of your jobs to use the values you specify.

For example, including the following commands in your user prolog specifies that all output from batch jobs will be routed to the NOS printer on the mainframe with the logical identifier (LID) of M02:

```
IF $job(mode) = 'BATCH' THEN  
  CHANGE_JOB_ATTRIBUTES du=dual_state dsrp='dc=pr,ec=a9,st=m02'  
IFEND
```

You could also have another CHAJA command in your job to override the values set in your user prolog.

NOTE

Your link attributes must be correct for the DSRP parameter to work correctly.

If you work from the host system (NOS or NOS/BE), you can route your NOS/VE batch job to NOS/VE and use the DC parameter on the ROUTE statement to specify output disposition. For example, using

```
ROUTE,MYJOB,DC=TO,ST=NVE.
```

on NOS will route file MYJOB to NOS/VE and cause the job's output to be placed in your NOS wait queue.

For NOS/BE, executing:

ROUTE,MYJOB,DC=IN,ST=NVE,TID=AB.

would route the file MYJOB to NOS/VE and cause the job's output to be sent to the user with a terminal id of AB.

6. How do I insure a unique status variable outside of a procedure?

First, use a common nomenclature for your status variable names. For example:

```
var##status
```

to signify a status variable.

You can then use the following command

```
var##status=$status(true)
```

which will work whether or not the status variable already exists.

This can be helpful for creating and using status variables both in groups of commands which are not inside of procedures and in FORTRAN programs which use SCLCMD.